2002

A doctor-patient communication tool (DPCT) Ryodoroku application on the web

Hongwei Bi

A DOCTOR-PATIENT COMMUNICATION TOOL (DPCT)

RYODOROKU APPLICATION ON THE WEB

A Project

Presented to the

Faculty of

California State University,

San Bernardino

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in

Computer Science

---

by

Hongwei Bi

December 2002

A DOCTOR-PATIENT COMMUNICATION TOOL (DPCT)

RYODOROKU APPLICATION ON THE WEB

A Project

Presented to the

Faculty of

California State University,

San Bernardino

---

by

Hongwei Bi

December 2002

Approved by:

[redacted]

Dr. David Turner, Chair, Computer Science — Date: Nov 27, 2002

[redacted]

Dr. Richard J. Botting, Computer Science

[redacted]

Dr. Arturo I. Concepcion, Computer Science

ABSTRACT

Doctor-Patient Communication Tool (DPCT) is a Web-based application that allows a Doctor to diagnosis patients remotely. The system is based on Ryodoraku, which is modern theory of diagnosis that has been refined from traditional Chinese medicine. There are many contemporary practitioners of Ryodoraku in the field of traditional medicine and its effectiveness has been supported by numerous case studies. Ryodoraku diagnosis relies on the collection of 24 electrical resistance measurements taken on various points around the hands, wrist, and feet. With DPCT, Patients input their symptoms and submit diagnostic data to their doctors remotely. Doctors give the instant diagnosis online. This system thus enables the regular collection of diagnostic data that can be used for a more comprehensive evaluation of patients' health condition. Additionally, the system provides useful functionality to both doctors and patients: doctors can maintain DPCT database easily; patients can view their case histories and update their personal information without time and location restriction.

ACKNOWLEDGMENTS

I give sincere thanks to my advisor, Dr. David Turner, who provided me invaluable guidance through the entire project. Under his patient, knowledgeable, and clear guidance, my project and my knowledge grew quickly. I also express sincere appreciation to my committee members, Dr. Richard J. Botting and Dr. Arturo I Concepcion, for their valuable advice. At the same time, I give the thanks to Dr. Shaolin Yeh, president of Yeh Center of Natural Medicine, Inc. and Dr. Dianmin Shen who gave me the primary design idea and valuable diagnosis data and guidance.

The support of the National Science Foundation under award 9810708 is gratefully acknowledged.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

INTRODUCTION

This project was suggested by Dr. Shaolin Yeh, an experienced Chinese Medicine Doctor, president of Yeh Center of Natural Medicine, Inc. and Dr. Dianmin Shen. They recommended me to design a Ryodoraku application on Windows platform. Considering that the Internet has been deployed broadly, and its application has huge potential, Ryodoraku application was designed as the Web-based application. The system has the prototype of becoming a useful tool in the field of acupuncture.

Traditionally, doctor and patient communication is based on making an appointment, and meeting in the doctor's office at an agreed time. This has a strict time limitation. More recently, some medical practitioners have been using a checklist of symptoms that the patient can use to get diagnosis and consultation in a more flexible time, such as "New Communication Tool, 2-COM", which was presented in MONTREUX, SWITZERLAND, in June 19, 2001 [11]. But all of these belong to static consultation. Nowadays, Web applications have been developed to allow doctor/patient consultations without requiring that the patient go to the doctor's office. The premier Web-based

doctor-patient communication tool was presented by Healinx Corporation in Alameda, CA, on April 3, 2000 [12]. It is a physician reference service, providing a group of doctors with a co-branded version of Healinx for their online doctor-patient messaging. People can get consultations online. Currently, all Web-based communication tools are limited to consultations without accompanying diagnosis. If the patients need to be diagnosed, they must go to the doctor's office or hospital. But if it is possible, why not communicate to the physician without going to the hospital?

This project provides a Web-based communication system for practitioners of Chinese medicine. People can be diagnosed more easily and frequently without going to the physician's office. However, online diagnosis is extremely difficult due to different people having various heath conditions. This project is based on the widely adopted RYODORAKU theory, detecting the Deficiency (Vacuity) or Sufficiency (Block) of people's Yuan points (Source points) to observe people's health situation. Once a person gets some disease, their body's balance is destroyed. Traditional Chinese medicine examines the yin and yang characteristics of 12 locations on the body, or "source points", resulting in 24 measurements. The 24

points are the main source for diagnosis. Qi is a Traditional Chinese Medicine (TCM) concept, it is hard to be translated, and it looks like energy. Neither Qi nor energy can be destroyed, only changed in their forms. Everything is composed of Qi; our bodies, the earth, water, sound, light. The Nei Jing - The Yellow Emperor's Classic of Internal Medicine (c.500BC) says "There is no place that Qi is not." [9]. Qi moves in people's body, the source points produce certain reflections (signals) according to different health situations. This project is to perform gathering of the reflecting signals from the 24 Source points, analyzing and managing these data, and diagnosing the different syndromes on the Web automatically. At the same time, the physician can also give detailed diagnosis remotely. Patients can get the diagnosis and consultation immediately and remotely. The project consists of two parts: Hardware design and Diagnosis Software System Design.

Hardware design - The Detection Device (DD) gathers the diagnosis signals and performs A/D conversion. A program runs in the client side to gather the DD output, and sends it to the Diagnosis Software System (DSS) running in the server.

Diagnosis Software System (DSS) design - The online diagnosis and communication includes user interface design, diagnosis software design, and communication system design. It will mainly be performed by ASP, Visual C++, Visual Basic, IIS, MySQL, ODBC, VBScript, and COM component. DSS analyzes the input data, performs the diagnosis and communication, and sends the results to the client.

CHAPTER TWO

REVIEW OF RELATED WORKS

Diagnosis Theory

The DPCT diagnosis system is based on Rydoraku theory. Rydoraku came from Traditional Chinese Medicine (TCM).

Traditional Chinese Medicine

TCM has thousands of years of history. It combines accumulated ancient knowledge with modern science. Due to its accuracy and flexibility, its diagnosis theory and methods have been accepted by acupuncturists broadly. The main theories of TCM are: Yin & Yang, The Five Elements, Qi, and Organ Functions. No matter how many theories and methods we have been used. The main diagnosis idea is to check people's circulation of Qi and the balance of Yin and Yang. The circulation checking is based on detection channels. In TCM anatomy, the internal organs of the body are all interconnected with one another by pathways called meridians, which are compared with Western ideas of the blood vessels and capillaries, or the nervous system with its centers and peripheral branching. The meridians spread out through the entire body connecting all the tissues and organs of the body binding it together as an organic unit.

They regulate normal functioning of the body, and diagnostically reflect pathology or illness. The most important and essential ones for the circulation of Qi, and for most therapeutic applications are the twelve Primary Meridians [8] (source points) in which will be used in DPCT.

The twelve Primary Meridians are: Shenmen (H_1), Daling (H_2), Taiyuan (H_3), Hegu (H_4*), Yangchi (H_5), Wanggu (H_6), Taibai (F_1), Taichong (F_2), Taixi (F_3), Jinggu (F_4), Qiuxu (F_5), Chongyang (F_6)(Fig 1).

* According to real world's case study, in Ryodoraku theory, by testing Yangqi as Figure 1 (H_4) instead of Hegu.

The twelve source points coordinate the body's 12 organs (Table 1). The relationship between the twelve organs and the source points is based on the five elements theory. The five elements are Water, Wood, Metal, Fire, and Earth.

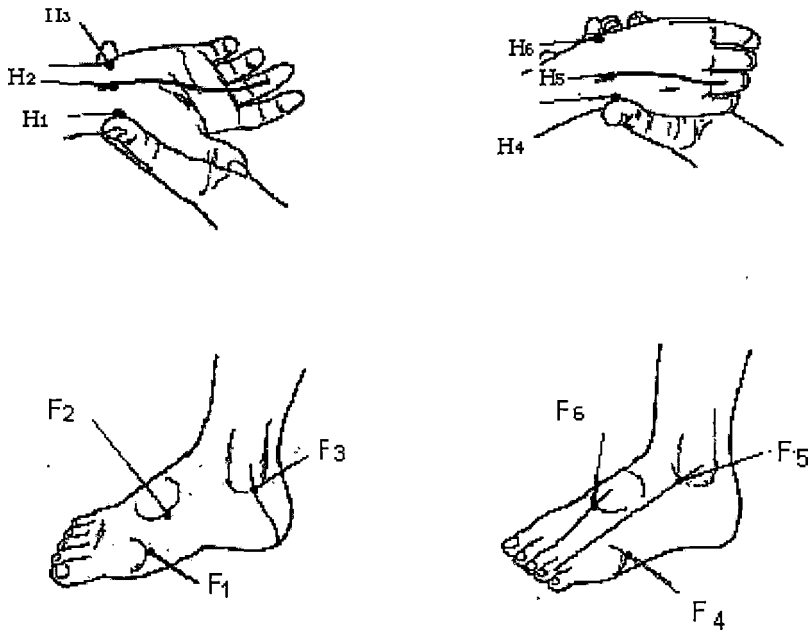


Figure 1. Twelve Detected Points

The relationship between the twelve organs and the source points is shown in Table 1.

Table 1. Relationship Between Organs and Points

Organs	Source Points
Lung	H1 (left, right)
Large Intestine	H6 (left, right)
Stomach	F6 (left, right)
Spleen	F1 (left, right)
Heart	H3 (left, right)
Small Intestine	H4 (left, right)
Bladder	F4 (left, right)
Kidney	F3 (left, right)
Pericardium	H2 (left, right)
Triple Heater	H5 (left, right)
Gall Bladder	F5 (left, right)
Liver	F2 (left, right)

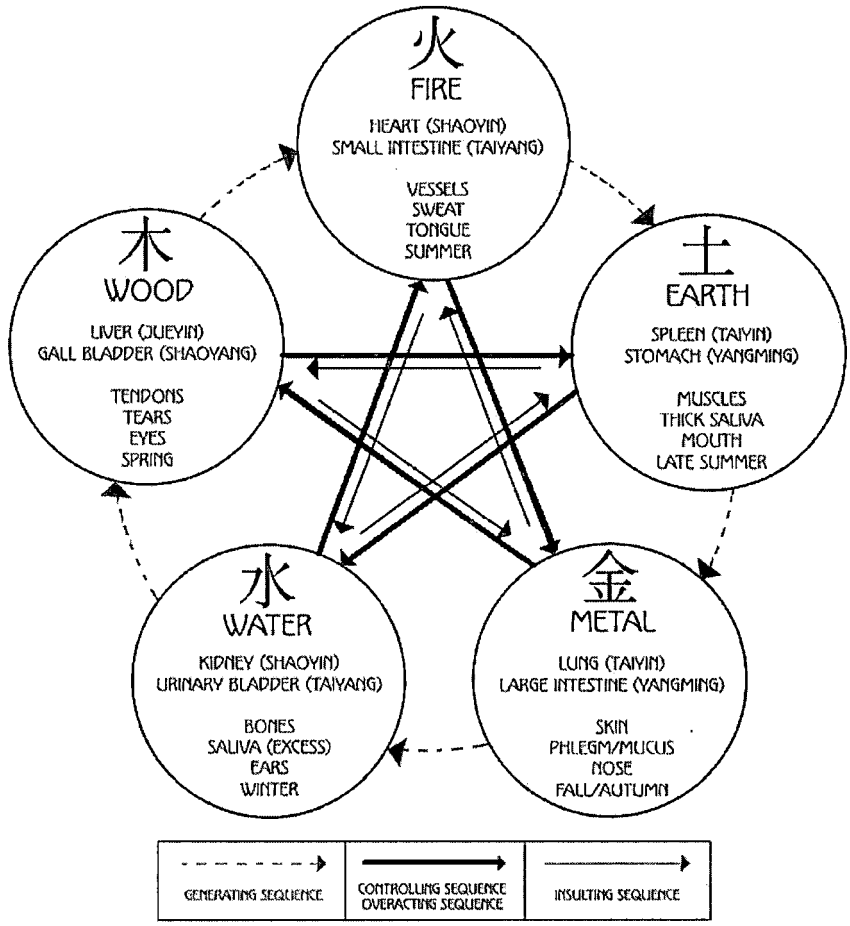


Figure 2. Five Elements Relationship

The flow of Qi through the circuit begins in the chest with the Lung channel of Taiyin (hand), and flows to the interior-exterior related Large Intestine channel of Yangming (hand). It then travels to the paired Stomach channel of Yangming (foot), and then returns to the interior-exterior related Spleen channel of Taiyin (foot). The Qi then ascends to the chest again before beginning a new circuit.

The circulation Figure as following:

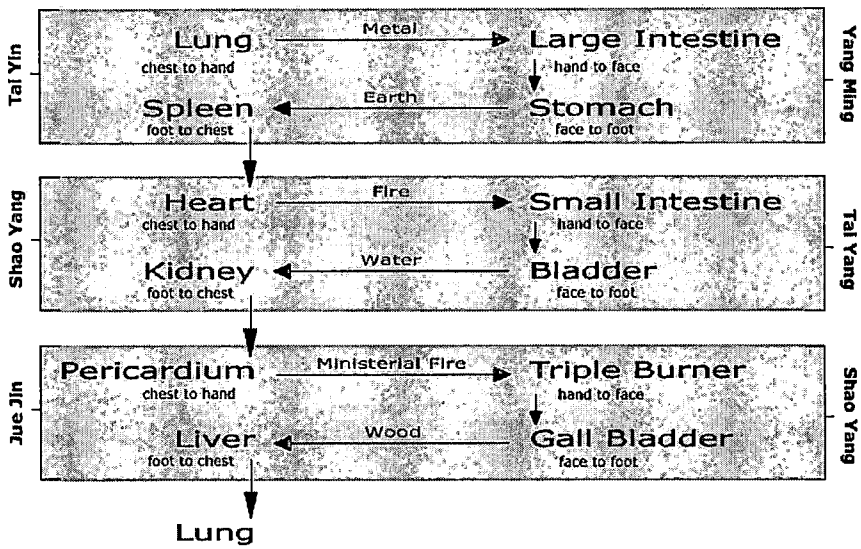


Figure 3. Inner Organs Circulation Diagram

Through both observation and Theorizing, numerous correspondences were discovered. When people's organs lose their balance, different diseases will occur. So by detecting the Meridian points, we can diagnose people's health situation.

Ryodoraku Theory

In 1951 Dr. Yoshio Nakatani presented his research and theory of RYODORAKU Acupuncture. Dr. Nakatani had found that there were a series of low electrical resistance points (or high electrical conductivity) running longitudinally up and down the body. When linked together these points closely matched the acupuncture meridians. Dr. Nakatani called these lines (or meridians)

"Ryodoraku" (*ryo* = good, *do`* is (electro) conductive, *raku* = line). The points along the Ryodoraku he named *Ryodoten.*

Dr. Nakatani was the first person to measure the electrical activity of acupuncture points and the first to formulate diagnostic and treatment criteria from these measurements. Numerous case histories had verified Ryodoraku theory's utilization and reliability.

Based on the TCM and Ryodoraku theory, DPCT detects the signals that come from body's source points as DPCT input. These data will be analyzed and manipulated during the diagnosis procedure.

Related Works

There are many online diagnosis systems, such as mentioned in Chapter 1. Recently, a more complete and advanced online diagnosis system emerged. It is the Health Balance LLC, located in Tusson, Arizona USA. This online diagnosis system based on a serial of questionnaires, including medical background such as: having been treated by Western Medicine and when? When did the condition begin? What was the related circumstance? What kinds of disease have or had been gotten? Patients also need to illustrate their perspiration, temperature, sleeping...

situations. The questionnaires are so completed that the patent needs to tell the system what color of his/her urine and how often. What about his and her sense organs situation and even his/her living hobbies? After inputting a bunch of required information, he/she can get the instant consultation. At the same time, a fee of $25 is charged. For more information, see [9].

The drawback of the system is that its accuracy depends on the patient's input. Even spending so much time to finish the questionnaires, sometimes it is hard to express our body's situation exactly.

Currently, all the online diagnosis systems isolate the patient body from the diagnosis system. Consequently, their accuracy is being challenged. This project absorbs the advantages of the other systems and adds physical diagnosis that based on RYODOROKU diagnosis theory. So this diagnosis system not only based on the patient's input, but also according to the real reflection from the patient body.

Ryodoraku hardware device has been made for many years, Such as Ryodoraku Health Monitor Series (RHMS) developed by Skylark Inc. [15]. It has the same data transmitting system as this project by using RS232. But Universal Serial Bus (USB) and Infrared (IR) transmission

are used for data transferring interface in RHMS. This is much more advanced than serial port interface like this product.

Significance of the Project

Ryodoraku theory has been used for over fifty year. Its application has been improved a lot during this time. The most advanced diagnosis system is designed on Windows. However, patients need to be diagnosed directly by the professional person.

The DPCT system has the advantages of the systems described in 2.2, and overcomes their main weaknesses. It is more convenient and deficient, because patients can remain at home and submit source point data without going to the clinic. These are benefits for both physicians and patients. Additionally, online communication and automatic diagnosis provides precious data for people to study and research. The system can also be used for education by providing access to a large number of case histories. Particularly, doctors in different parts of the world can use this communication system for consultation at the same time.

Limitations of the Project

Considering that ASP has great convenience and integration with Microsoft IIS, this project was developed on Windows 2000 professional, in which IIS is embedded. This restricts the DPCT server to the Windows platform. To port the application to a different platform, please consult Chili!ASP or Halcyon Software's InstantASP. Refer to Chapter 6 for details.

This project is constrained in that each patient can only store one case history per day. Suppose the patient sends data sent twice in one day to DPCT Web server, then latest one will erase the former one.

Constrained by the hardware specification, the application has a strict temperature requirement. The client detection device should be operated in the range 0 ~ +70 C.

CHAPTER THREE

REQUIREMENT SPECIFICATION

Introduction

Purpose

The purpose of this document is to provide a concise and clear "Software Requirement Specification" for Doctor-Patient Communication Tool (DPCT) project. This document will comply with the basis agreement between the customer and the supplier. Hence to provide the prototype of online diagnosis with the diagnostic signal gathered from both physical diagnostic information and patient input symptoms and server as a basis for future enhancement.

The intended audience for DPCT will include:

Physicians: Giving the instant online diagnosis.

Patients: Attending to get instant diagnosis online.

Scope

The DPCT provides a Web-based communication system for practitioners of Chinese medicine. People can be diagnosed online without time and location constrains. Patients can diagnose themselves as many as they want and check case histories anytime. Physicians can maintain the DPCT's database without knowing any database knowledge. At the same time, user easily updates their personal

information. Therefore, this product provides clear benefit for both patients and physicians.

DPCT will be produced by following hardware and software.

Hardware:

1. Circuit board
2. Transformer 220V/5V/12V
3. Standard PC transceiver
4. A/D converter
5. Capacitors
6. Resistant
7. Switch
8. Connector
9. Wire

Software:

Server side:

1. Microsoft win32 operating system
2. Internet Information Servers (IIS 5.0)
3. MySQL server
4. Web browser which ASP supported
5. ASP 3.0 or above
6. ADO 2.5 or MDAC 2.5
7. ODBC Connectivity
8. 64MB RAM

Client side:

1. Microsoft windows operating system

2. Web browser which ASP and VBScript supported

Definitions, Acronyms, and Abbreviations

Refer to APPENDIX B.

References

Refer to REFERENCES of DPCT system.

Overview

The remainder of this document defines the functions and specific requirements of DPCT in a format consistent with the IEEE Std 830-1998 SRS format [14].

Overall Description

Product Perspective

- System Interfaces. The DPCT system is a completed independent product. All hardware and software components have been prototyped for a complete working system.

  This product consists of four tiers: hardware tier, presentation tier, diagnosis tier, and database tier.

  The interface between patient and Detection Device (DD) are connectors in which gathering the patient body's signals. The interface between DD

and computer is a serial port line sending converted signals to computer. The data transfer through FTP from presentation tier to diagnosis tier. ODBC and ADO are designed as database connectivity.

Deployment Diagram:

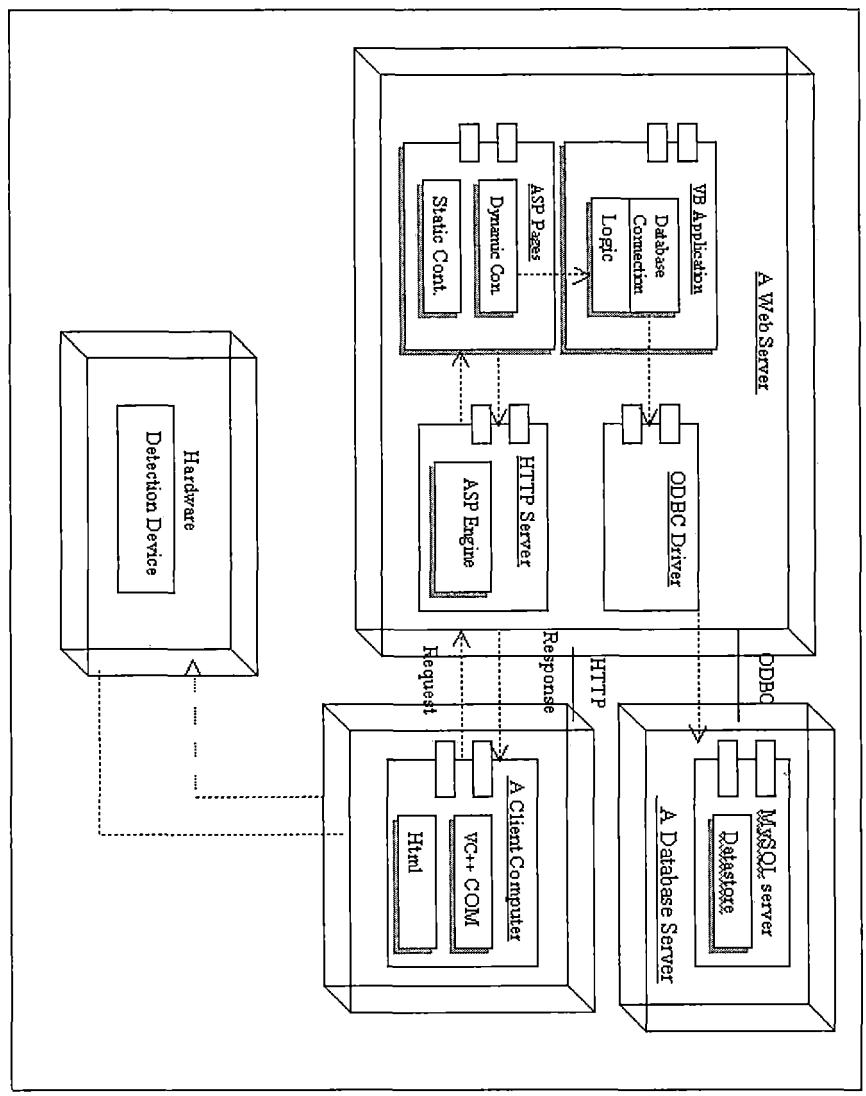


Figure 4. Deployment Diagram

- User Interfaces. The DPCT system mainly contains two kinds of users: The physician and the patient. All the interactions between users and the system are performed through ASP pages. But the physician and the patient have different interactive pages. The first page for physician is as Figure 30. The first page for patient is as Figure 31. For detailed user Interfaces refer to "External Interfaces".

  All DPCT system's API screens are adjusted. People can also access DPCT Web system through general Web browser such as Internet Explorer or Netscape Navigator.

  When patients diagnosis themselves, they are supposed with basic TCM knowledge in order to increasing the diagnosis precise.

- Hardware Interfaces. Port COM 1 was designed as communication Port. A standard compatible data interface part RS232 was used in the hardware.

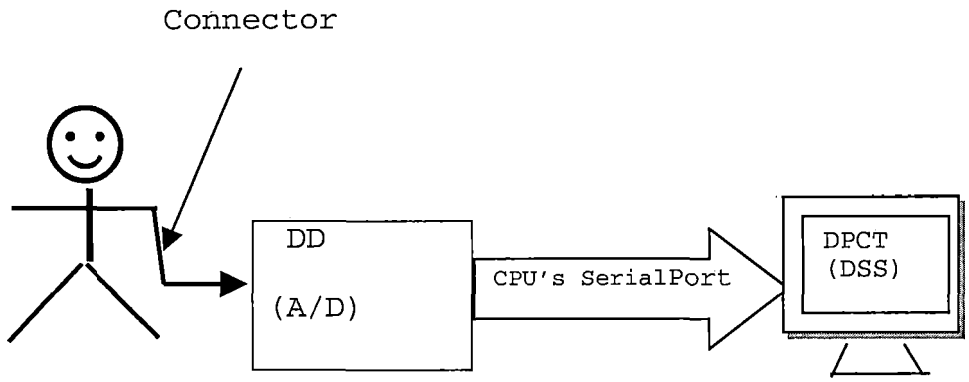


Figure 5. Overall Chart

- Software Interfaces. This system was developed on Windows 2000 platform with IIS 5.0 and FTP server embedded automatically.

  Presentation tier is standalone software with coded in Visual Basic. It was developed on VB6.0. The Web pages are developed in ASP 3.0 with VBScript and JavaScript. The database connectivity is through ADO 2.5 and ODBC. Source codes refer APPENDIX A.

- Communication Interfaces. Data communication protocol is under TCP/IP. Physician and patient communication is through Web page. Patients can notify physician by email or telephone.

- Memory Constraints. For server side 64 RAM is recommended. No specified memory constraint for client side.

- Operations. The product can distinguish physician and patient status. So the operation limitation has been performed in software. Therefore no specified limitation and requirement for user.
- Site Adaptation Requirements. W32 system with Internet Information Server (IIS) and FTP installed are available for server side. For client side, Web browser installed with no special requirement granted by ASP characteristics.

Product Functions

- User Case Diagram

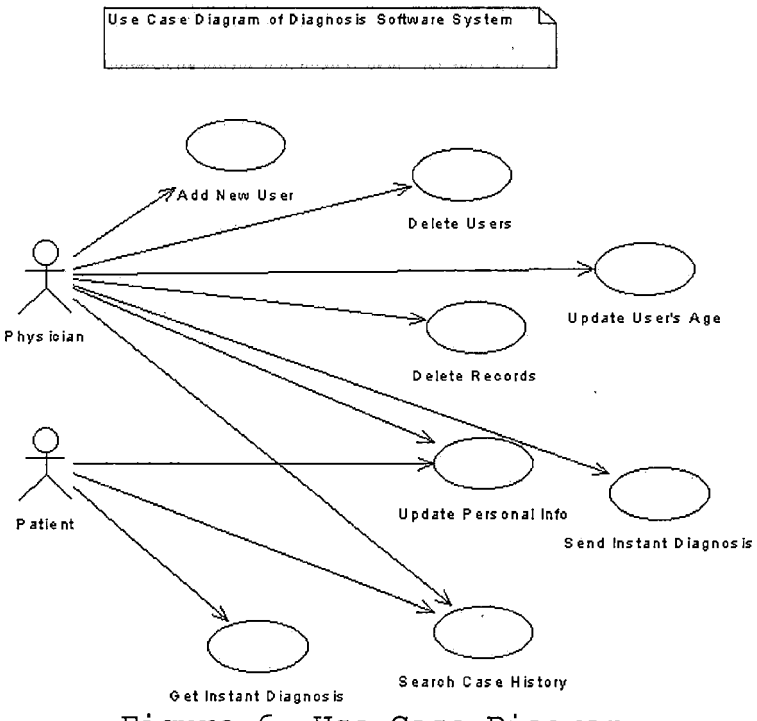


Figure 6. Use Case Diagram

User Characteristics

The product users DPCT will include any people who intend to diagnose his health. These individuals need to know the location of these diagnosis points exactly in order to increase the diagnosis correctness and preciousness. The basic computer operations and vocabulary ability are required.

Constraints

The hardware environment temperature is 0-70ºC. All the power supplies for the hardware are 5V and 13V under safety Voltages.

Assumptions and Dependencies

This product is embedded on Windows OS. Intend to use or develop on other platforms. Some configuration and codes need to be change or updated.

Specific Requirements

External Interfaces

- ExecForm. The Exec form will be displayed whenever people click the DPCT shortcut icon. This page embeds the DPCT Web default home page. Forward and back are used to navigate between application Web pages.

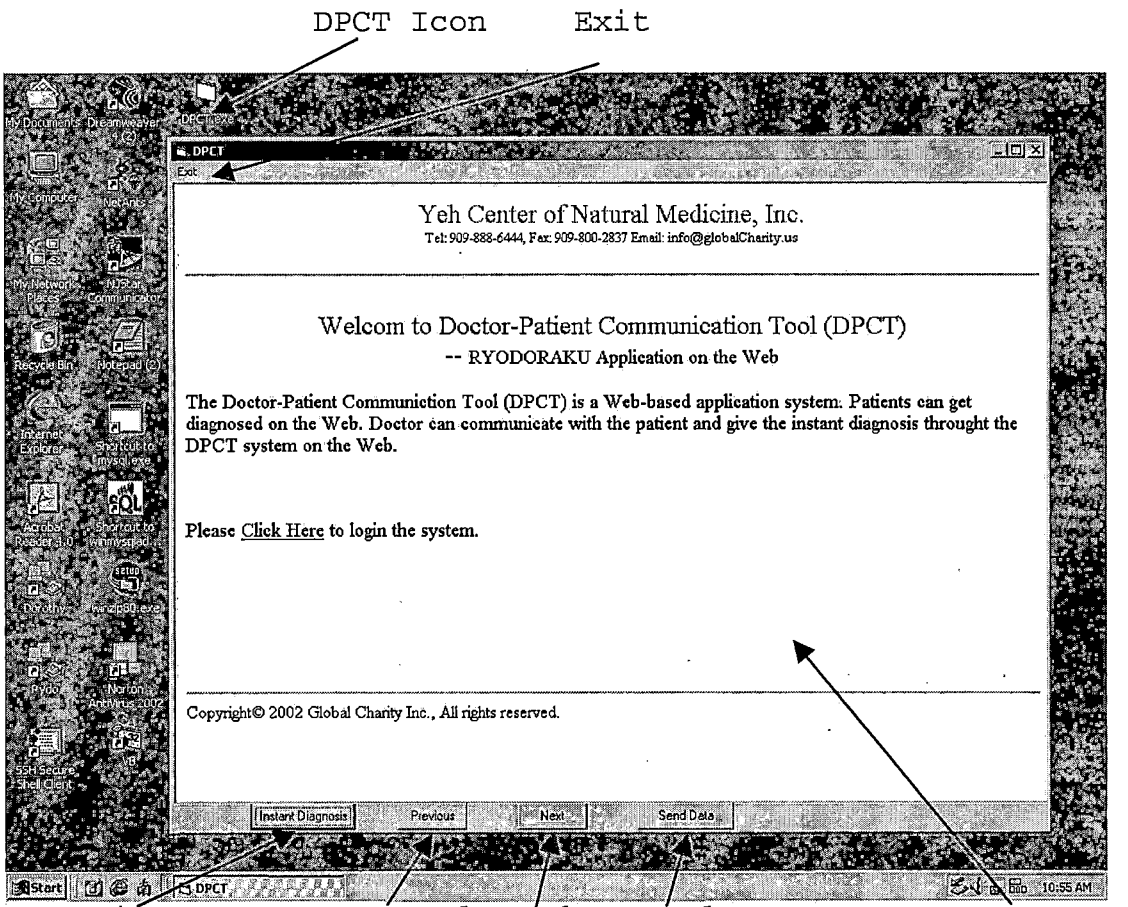


Figure 7. ExecForm

- Point 1. When the user clicks the Instant Diagnosis button, DPCT begins to Communicate to hardware for receiving and converting data.

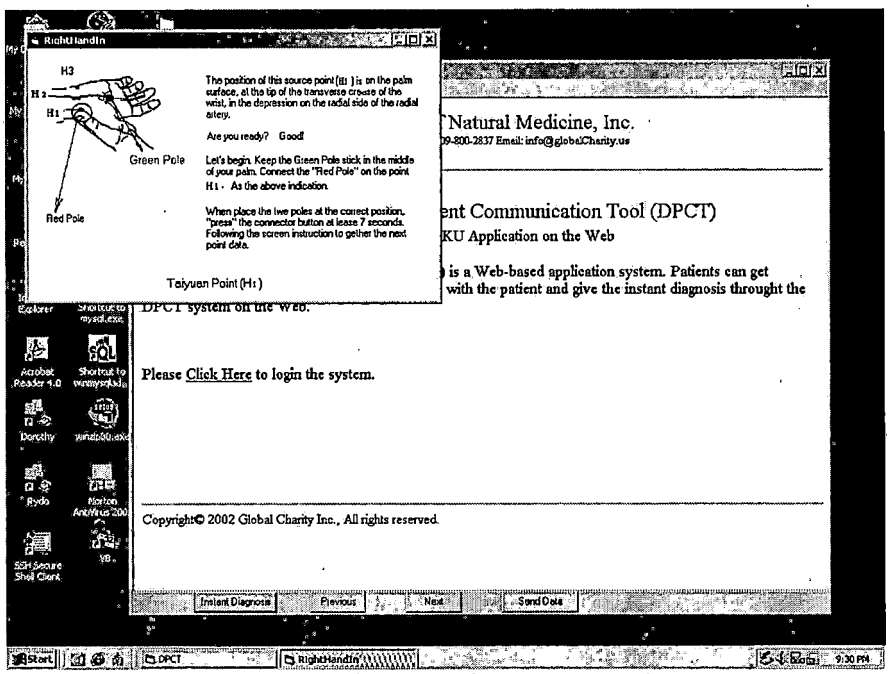


Figure 8. RightTaiyuan (Point 1)

The user puts one pole of the DPCT hardware connector (Green Pole) in the patient's right hand. The patient also can diagnose himself personally. The user then places the Red pole on the correct point to be diagnosed. People can follow the screen's direction as shown in Figure 9. The user places the Red pole on the Taiyuan point (H_1) indicated by the Red circle.

The position of this source point is on the palm surface, at the tip of the transverse crease of the wrist, in the depression on the radial side of the radial artery.

Once the user places the connector in the

right places, he/she shall press the connector's button for at least 7 seconds. At this time, the detected data has been sent to the PC, the screen will change to the next point direction as shown in Figure 10.

- Point 2. Change the "Red Pole" to left hand. As Figure.10 directions, testing Yin source point Taiyuan (H_7). The procedure is as before keeping the "Green Pole" sticking in your right hand's palm. The point location refers to point 1.The user should use the points carefully. When pressing the connection button, the user should presses the button for 7 seconds. Then the screen as Figure. 11 will be displayed.

Diagnosis Point

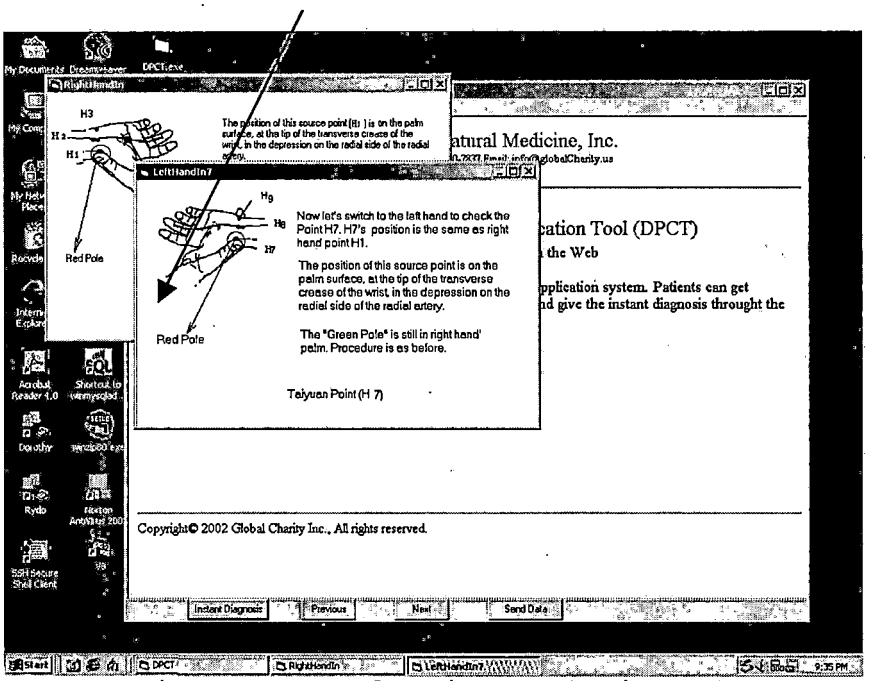


Figure 9. LeftTaiyuan (Point 2)

- Point 3. Following the directions shown in Figure 11, people begin to test the third point, Daling (H_2). During the entire testing, the user keeps the "Green" pole on the palm of the right hand. This point's location is at midpoint of the transverse crease of wrist between the tendons of m. palmaris longus and m. flexor carpi radialis.

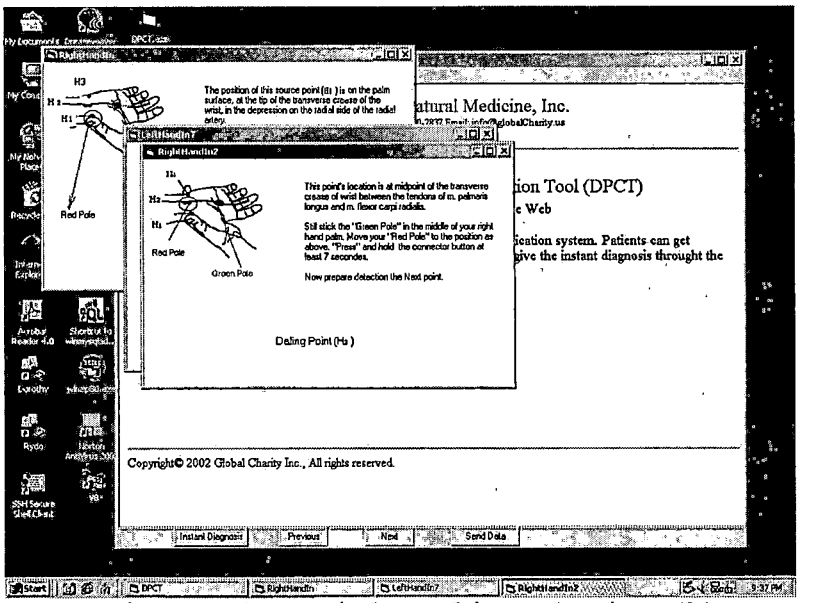


Figure 10. RightDaling (Point 3)

- Point 4. Testing Left Daling (H_8). Location is the same as point 3.

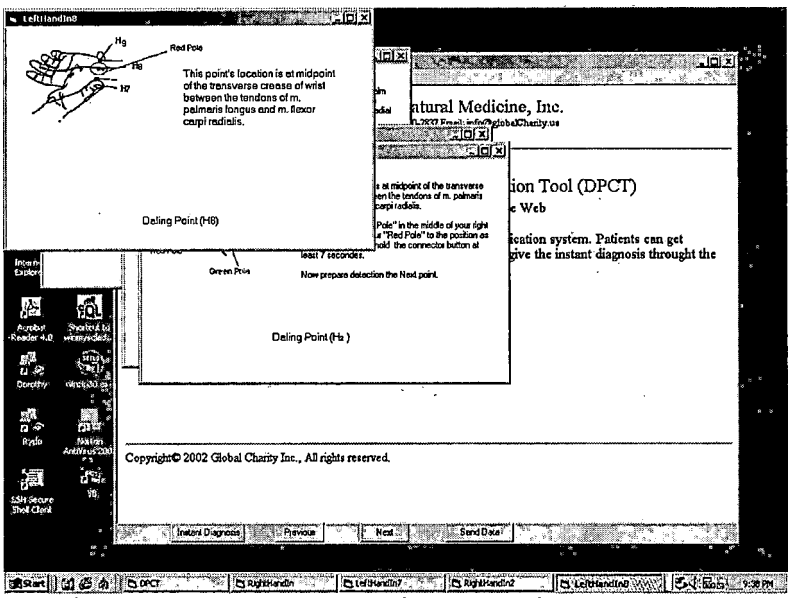


Figure 11. LeftDaling (Point 4)

- Point 5. The procedure of testing point 5, Shenmen (H_3) is similar to the procedure for points 1 and 2. Its location is on the lunar side of the wrist, on the posterior border of the pisiform bone, in the depression at the radial side of the tendon of m. flexor capitularies.

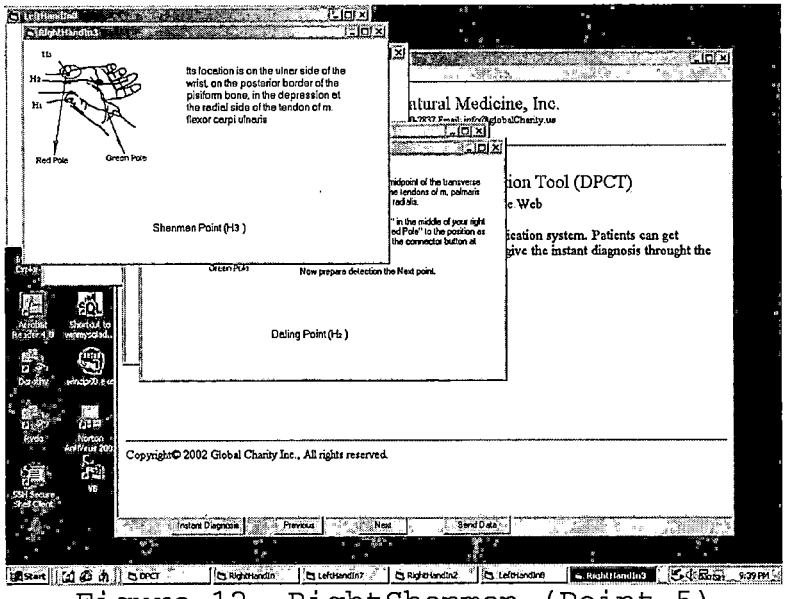


Figure 12. RightShenmen (Point 5)

- Point 6. Testing left Shenmen (H_9). The position refers to point 5.

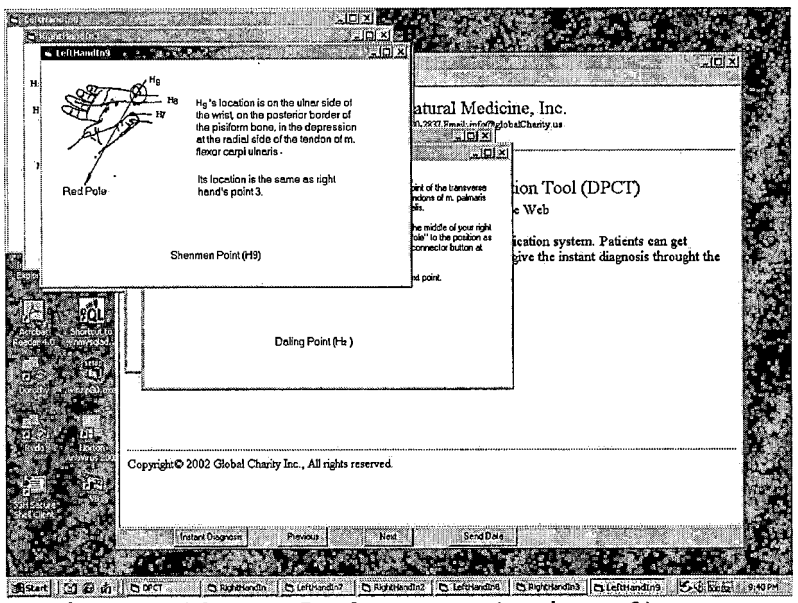


Figure 13. LeftShenmen (Point 6)

- Point 7. After testing right hand's Yin source points, begin to test the Yang source points. The directions guide the user to test the Wangu point (H_4). The procedure for testing the right outer hand follows the same pattern as the inner hand.

  Wangu point's location is at the lunar side of the border of the palm in the depression between the base of the 5^{th} metacarpal bone and the trigonal bone.

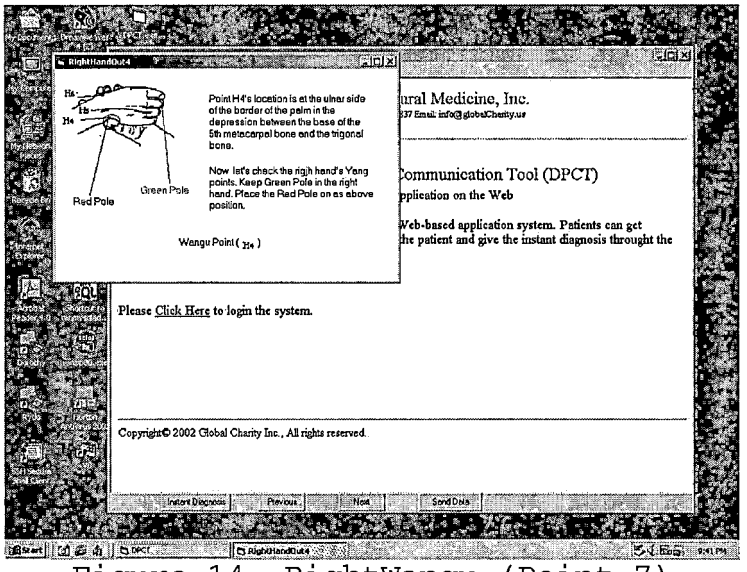


Figure 14. RightWangu (Point 7)

- Point 8. Begin to test left Yang source points. Refer to Point 7's position, place "green pole" in left Wangu (H_{10}). The procedure is as before.

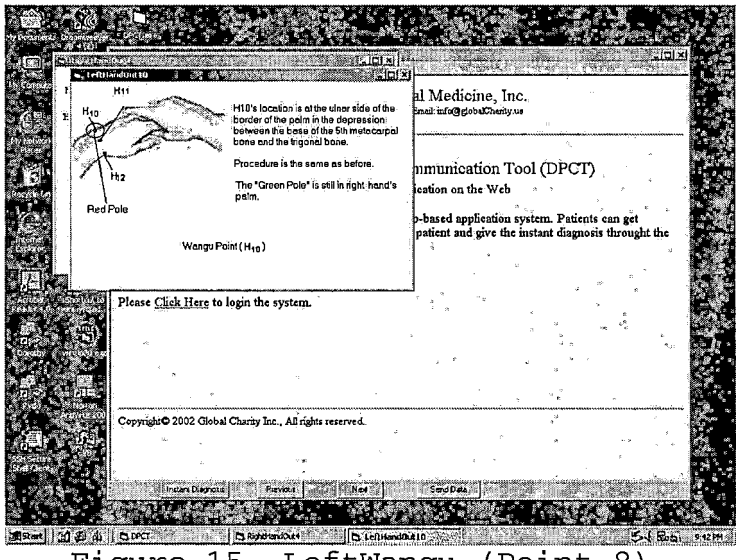


Figure 15. LeftWangu (Point 8)

- Point 9. As directed, test the source point Yangchi (H_5). Its location is in a depression of the transverse crease of dorsum of wrist between the m. extensor digitorum communis and m. extensor digit quinti proprius. [1]

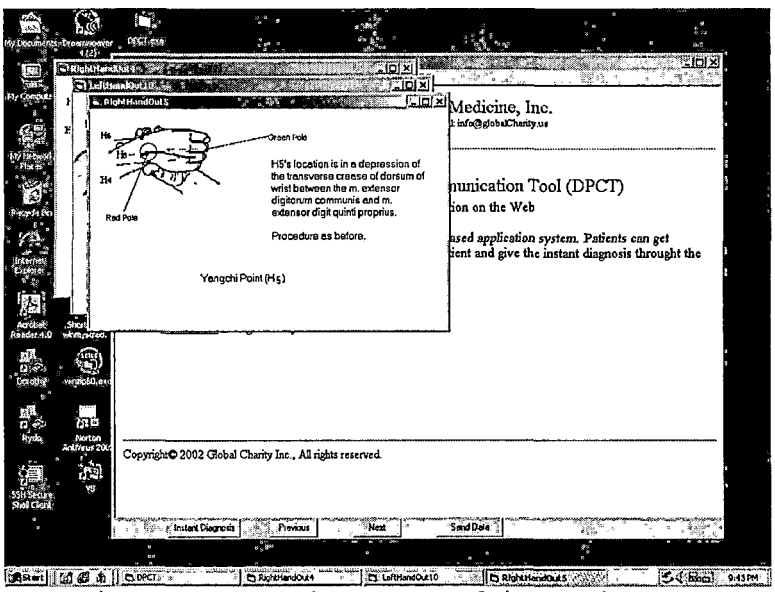


Figure 16. RightYangchi (Point 9)

- Point 10. Testing left hand Yangchi (H_{11}) point Location is the same as point 9.

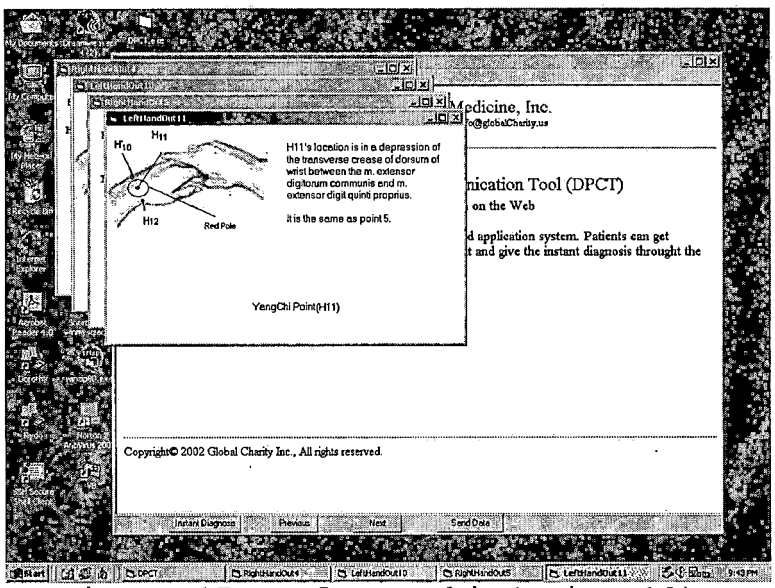


Figure 17. LeftYangchi (Point 10)

- Point 11. Following the directions on the screen to test the right hand Yangxi (H_6) point. Its location is on the radial side of the back of the wrist. When the thumb is titled upward, it is in the hollow between the tendons of m. extensor policis brevis.

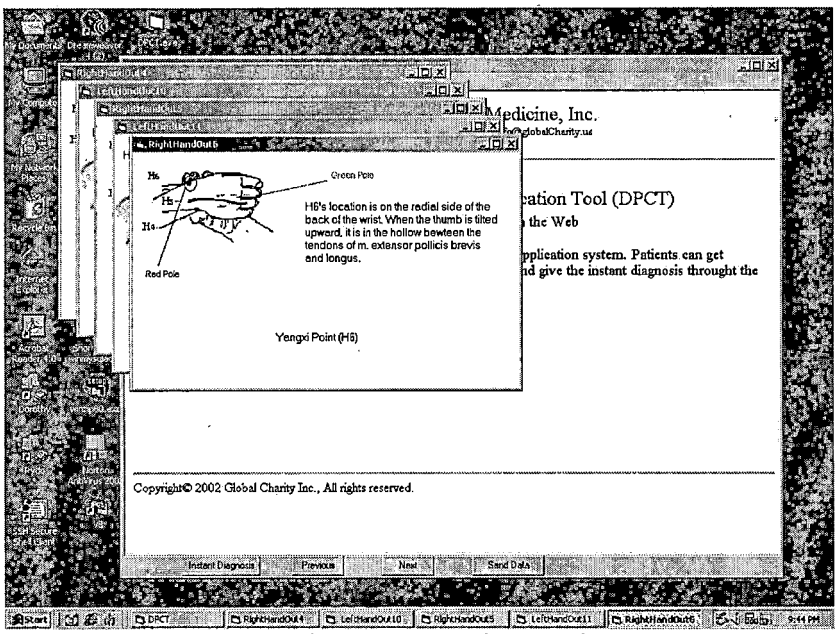


Figure 18. RightYangxi (Point 11)

- Point 12. Testing left hand Yangxi (H_{12}). This is the last point of hands' Source points. After testing this point, switch to the right foot. Its location is the same as point 11 the procedure is as before.

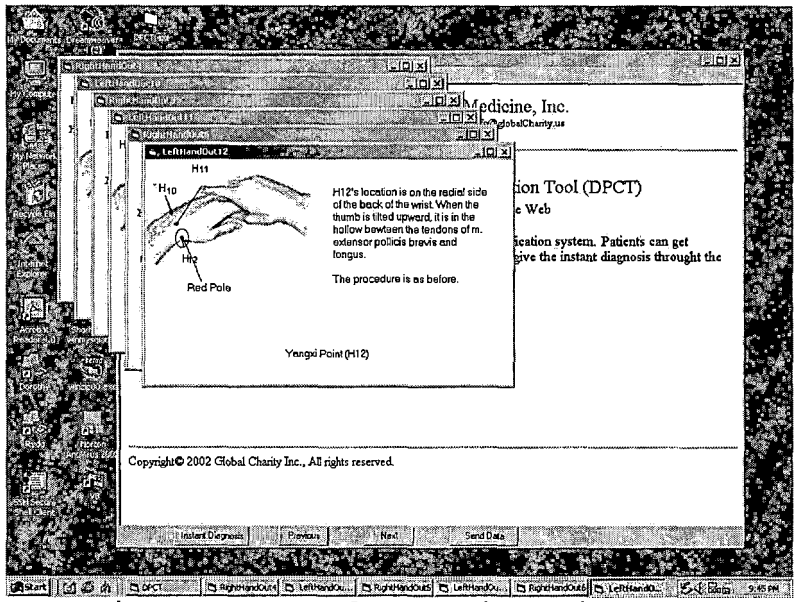


Figure 19. LeftYangxi (Point 12)

- Point 13. Testing Yin foot source point Taibai (F_1). Its location is at the medial aspect of foot, posterior and inferior to the head of the first metatarsal bone, at the junction of the "red and white" skin. [1]

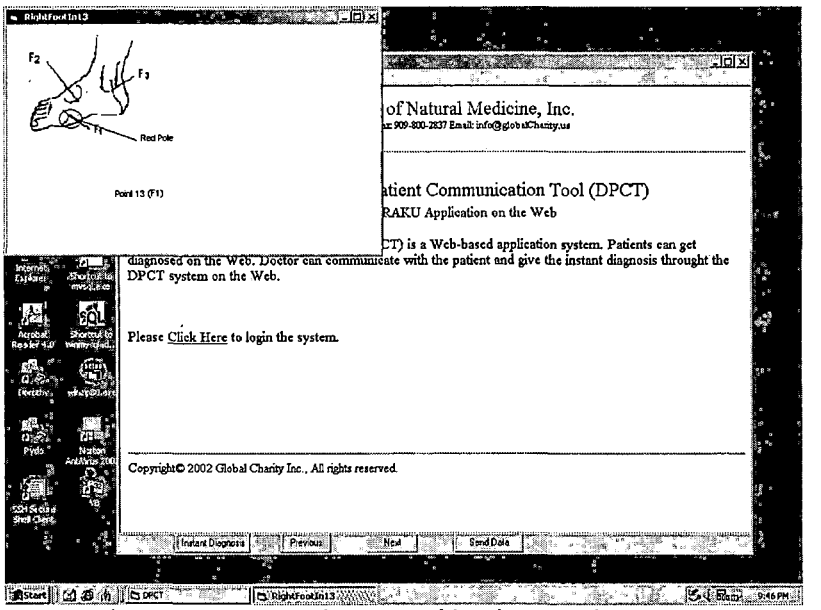


Figure 20. RightTaibai (Point 13)

- Point 14. Testing left foot Yin source point Taibai (F_7). Its location is the same as Point 13.

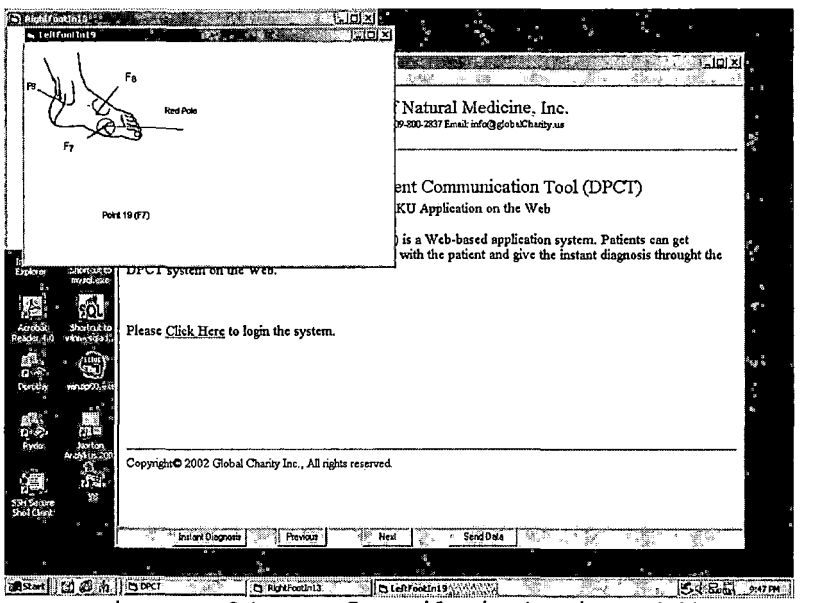


Figure 21. LeftTaibai (Point 14)

- Point 15. Testing Taichong(F_2). This position is 1 cun anterior to the medial malleolus, in the depression medial to the tendon of m. tibialis anticus. [1]

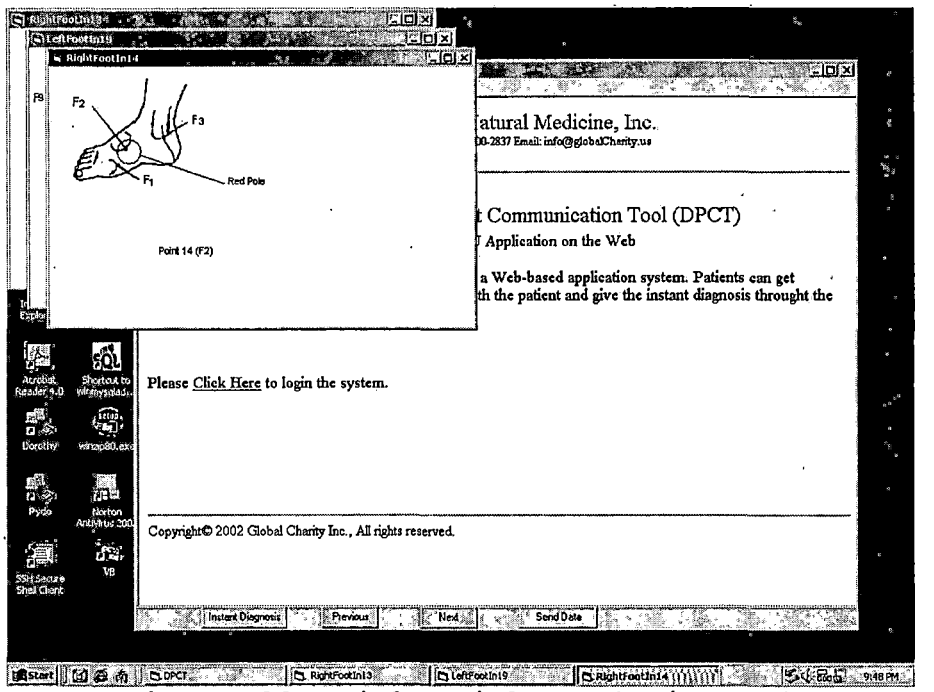


Figure 22. RightTaichong (Point 15)

- Point 16. Testing Taichong of left foot (F_8). Its location is the same as point 15.

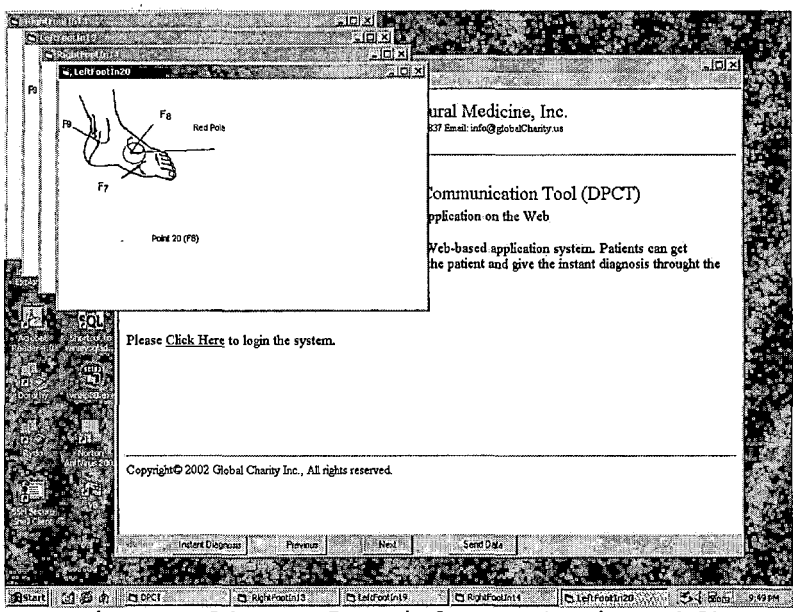


Figure 23. LeftTaichong (Point 16)

- Point 17. This page displays the Point 15 Taixi (F_3)'s It is in midway between the tip of medial malleolus and tendo calcaneus. [1]

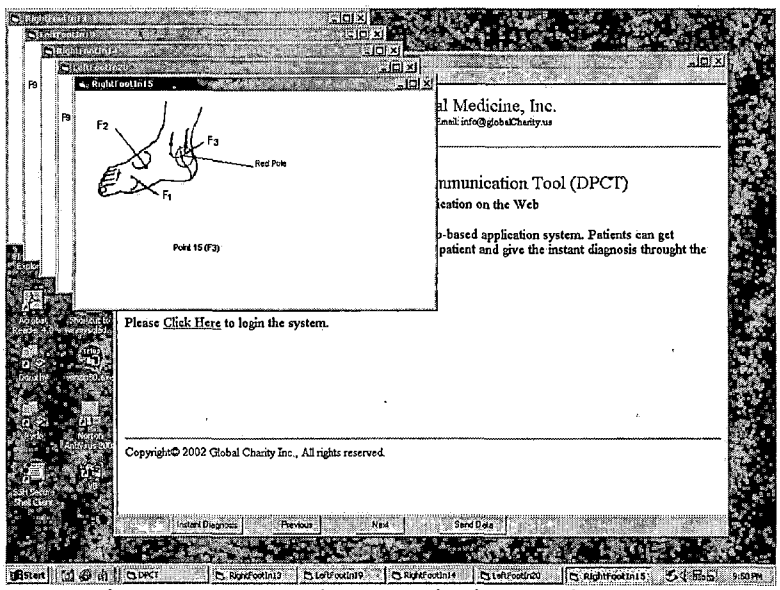


Figure 24. RightTaixi (Point 17)

- Point 18. The left Taixi point's position is the same as point 17. When testing feet points, still keep the "red pole" in right hand palm.

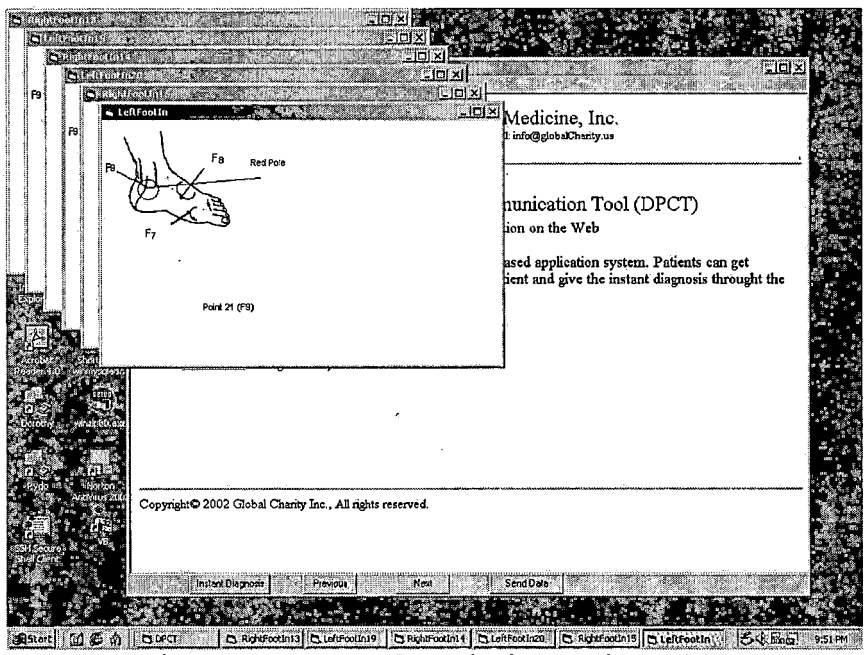


Figure 25. LeftTaixi (Point 18)

- Point 19. Testing Jinggu(F_4). Location is at below the tuberosity of the 5th metatarsal bone, at the junctions of the "redand white" skin. [1]

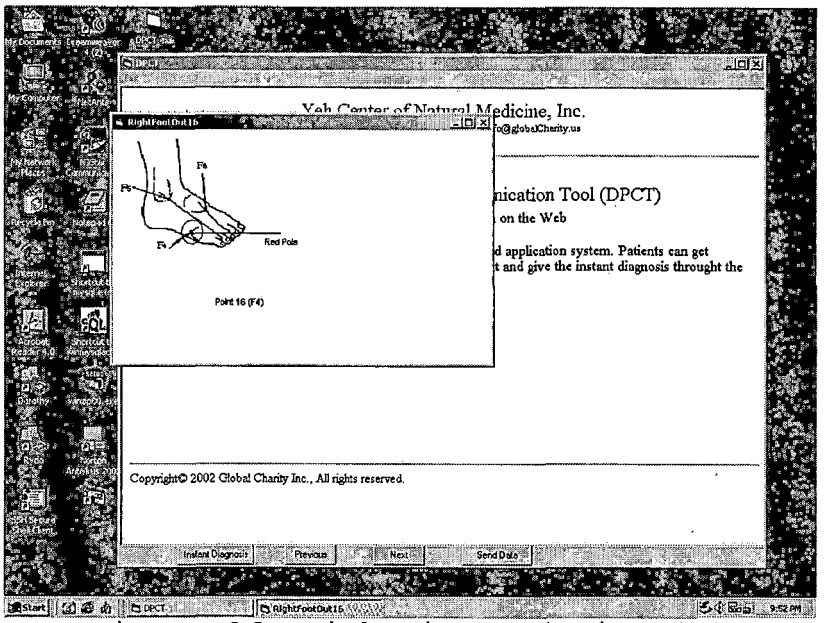


Figure 26. RightJinggu (Point 19)

- Point 20. Testing left Jingu (F_{10}). As point 19.

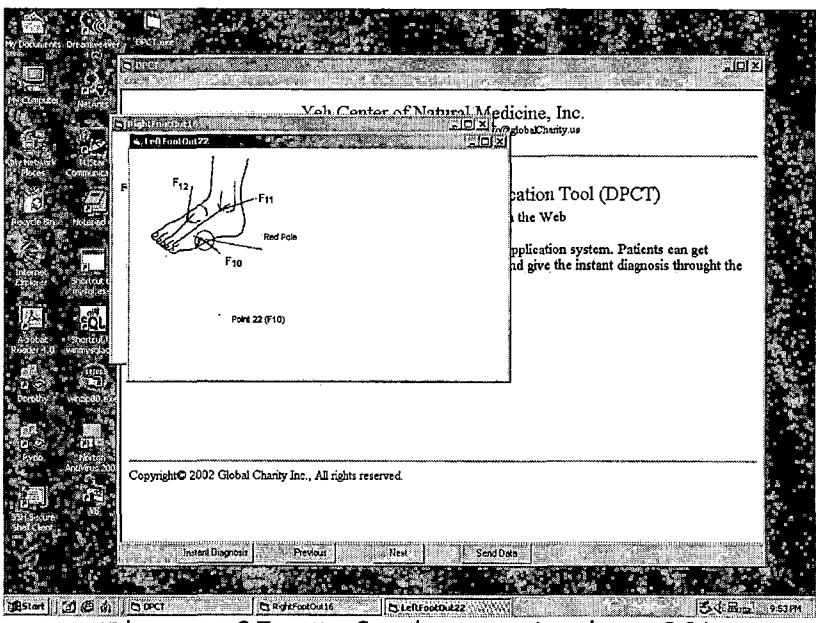


Figure 27. LeftJinggu (Point 20)

- Point 21. Testing Qiuxu (F_5). Its location is at anterior and inferior to the external malleolus, in the depression on the lateral side of the tendon of m. extensor digitorum longus. [1]

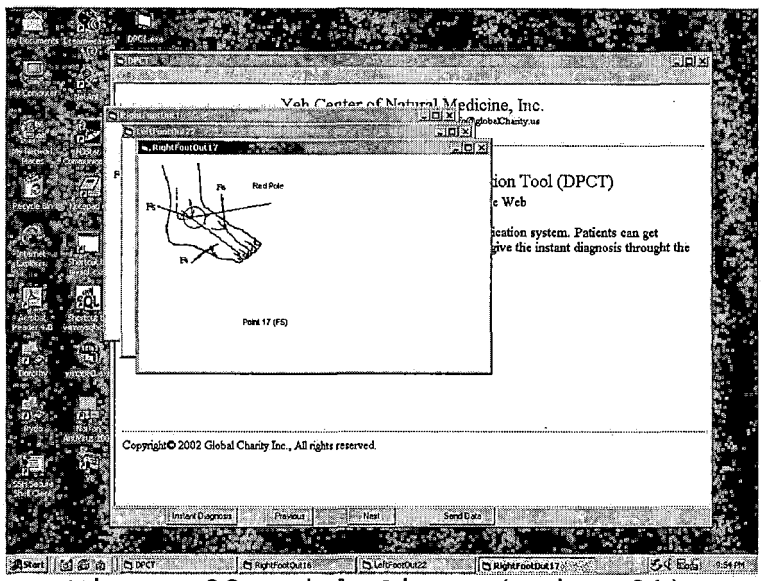


Figure 28. RightQiuxu (Point 21)

- Point 22. Testing left Qiuxu (F_{11}). The position is the same as point 21.

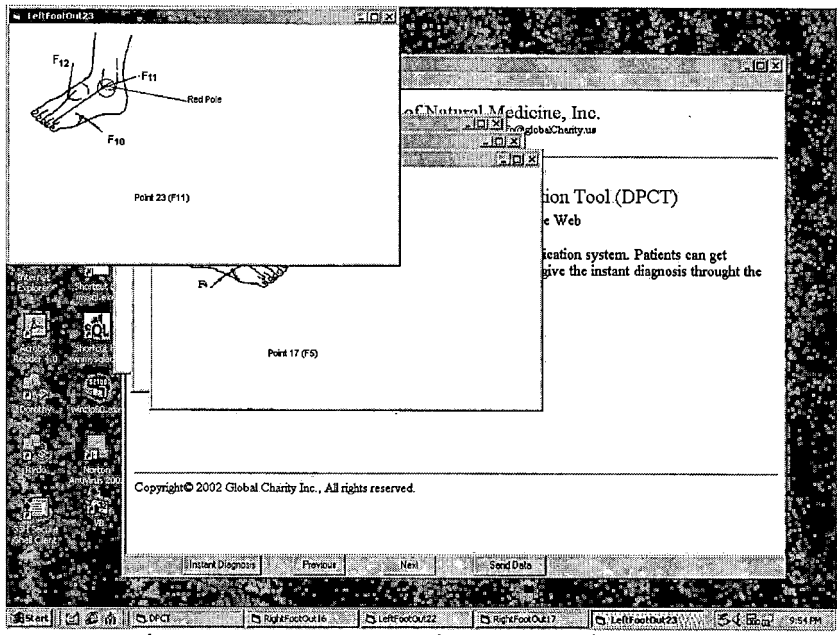


Figure 29. LeftQiuxu (Point 22)

- Point 23. Testing right Chongyang (F_6). Location is 1.5 cun distal to Jiexi, at the highext spot of dorsum of foot, artery can be palpated. [1]

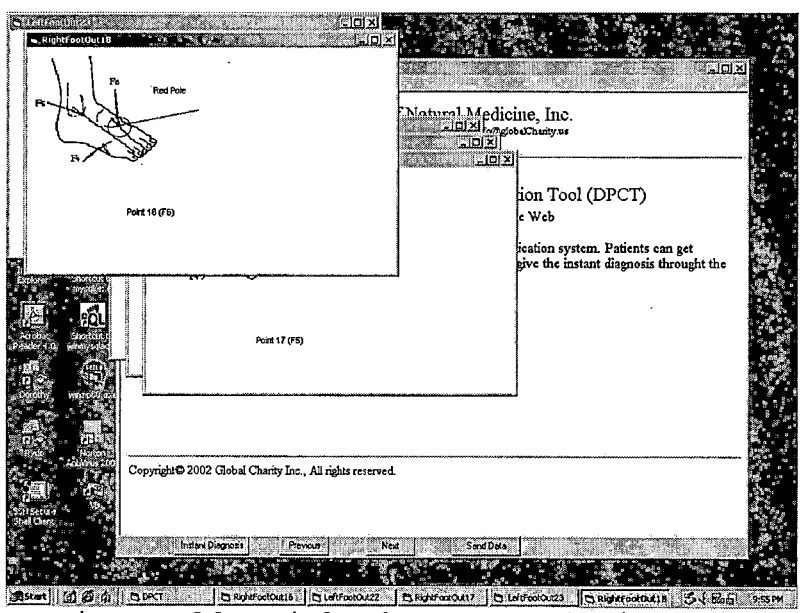


Figure 30. RightChongyang (Point 23)

- Point 24. This is the last testing point Chongyang (F_{12}). As point 23. As soon as the detection finishes, the data has been stored in a file. By clicking the Send button, the data and the related image will be sent to DPCT server for further manipulation.

Login in

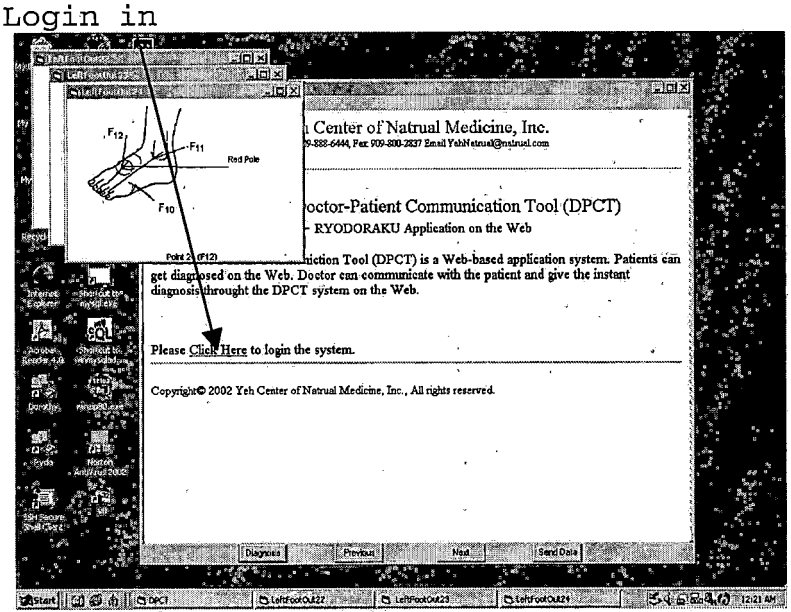


Figure 31. LeftChongyang (Point 24)

After sending the data, people can login to DPCT and check the diagnosis result.

- Login Page. After clicking the login link, the Login page will be displayed. After the user enters his/her email Address and password, he/she can navigate the DPCT Web pages. Doctors and patients access the DPCT Web site through the

same entry page, but they have different functionality.

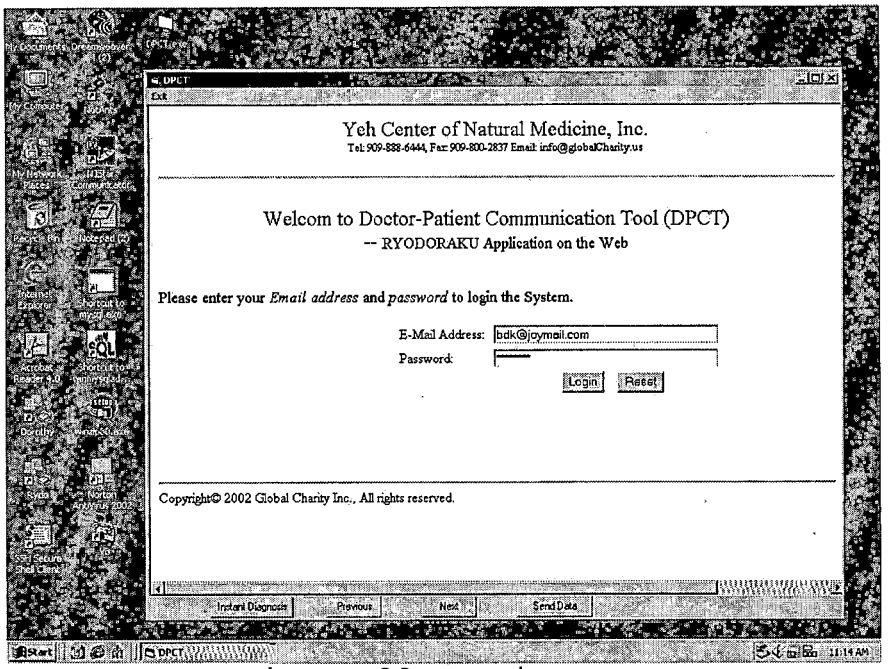


Figure 32. Login Page

The doctor can add and delete users, modify the diagnosis results, update user information, and view patient case histories. Patients can change passwords, update personal information, and search their own case histories.

Doctors and patients obtain different response pages according to their status.

- Sub_Homepage. Sub_Homepage is the second page when the doctor login the DPCT system. It provides logic search button that search patients records.

Also, through this page, doctors can update their personal information, add new user, and update users' age functionalities.

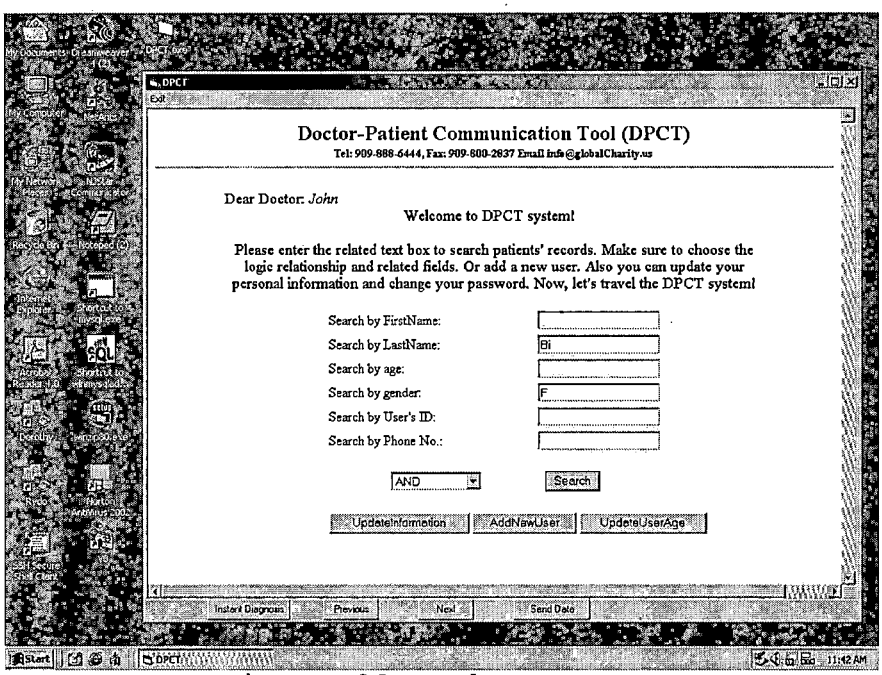


Figure 33. Sub_Homepage

This page allows doctors to search patient's diagnosis history by inputting the correspondent text fields. If people try to do some logic search, make sure the Select field is in the correct relationship.

After clicking UpdateInformation button, Figure 36 will be displayed. User can update their information anytime. During surfing the DPCT system, people can exit the DPCT system anytime by

clicking "Exit" menu bar. The doctor can navigate to the previous page and the next page by pressing the outer form's buttons: Previous and Next.

The Update Users's Age button is used to add one-year to all users in the DPCT database at every beginning of New Year.

- PatientLogin Page. This page is the second page when patient login in the DPCT system. People can check the instant diagnosis result by clicking the Diagnosis button. The diagnosis result refers to Figure 35.This page also provides update personal information link and search case histories link.

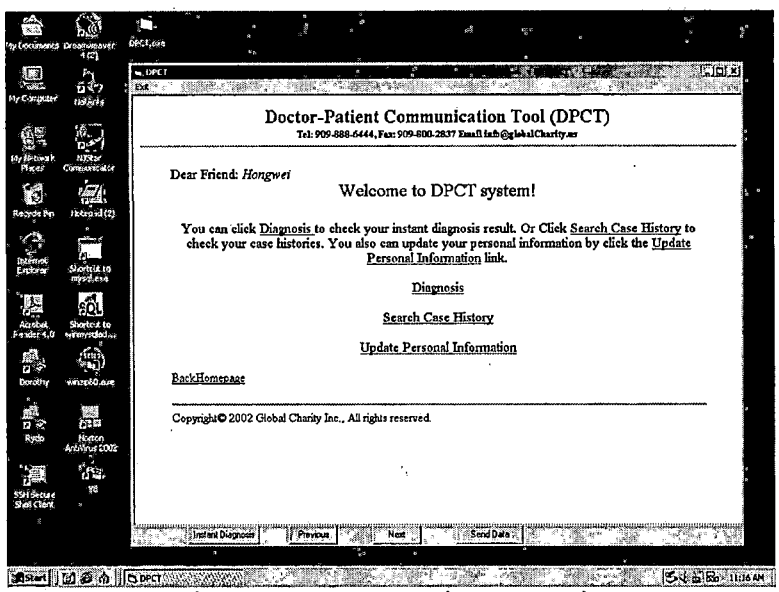


Figure 34. PatientLogin

- Diagnosis Page. This page shows the current diagnosis result. It is a scroll page.

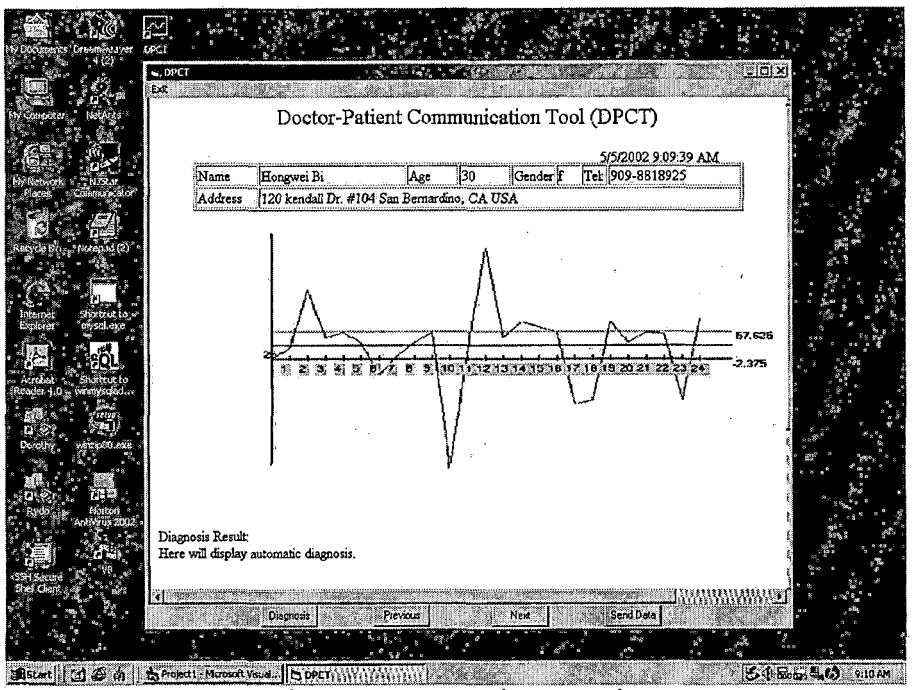


Figure 35. Diagnosis Page

From this page, the patient can automatically get the diagnosis result. Patients can type in their symptoms. Doctors can write a detailed diagnosis.

- UpdateInfo Page. When users want to update their personal information, they can change the corresponding text fields. This page is accessed through Sub_Homepage or patientLogin page.

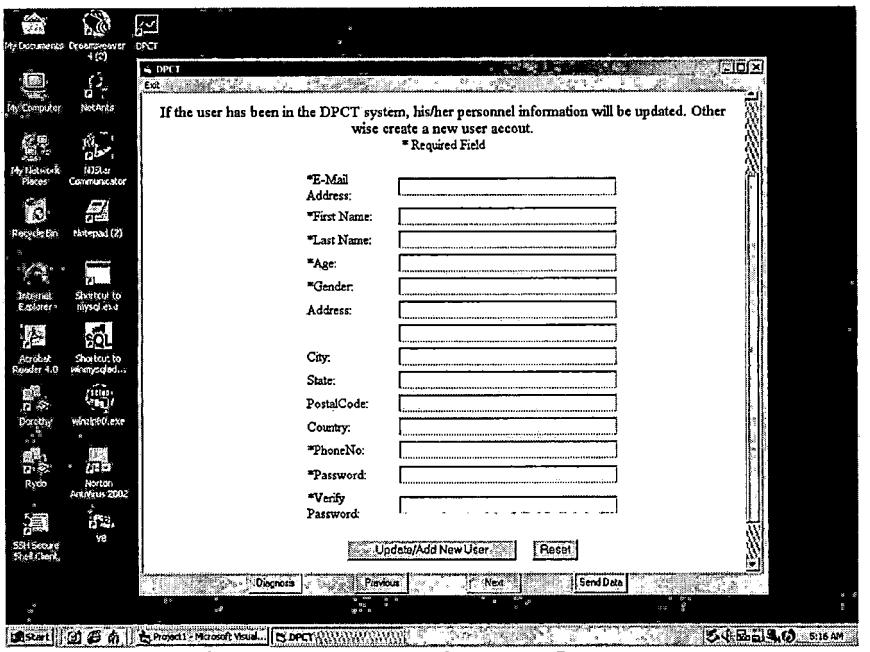


Figure 36. UpdateInfo Page

- SearchByFName Page. In Sub_Homepage, doctors can view case histories. Figure.37 illustrates the search result by Checking the First Name is Hongwei. Each item has a Delete button used for deleting the related user, but only the physician has this privilege. When the physician deletes a user, all related information stored in the DPCT database will be deleted. By clicking the Record button, all records about the patient will display as Figure 38. By selecting related the check box, the detailed diagnosis will display.

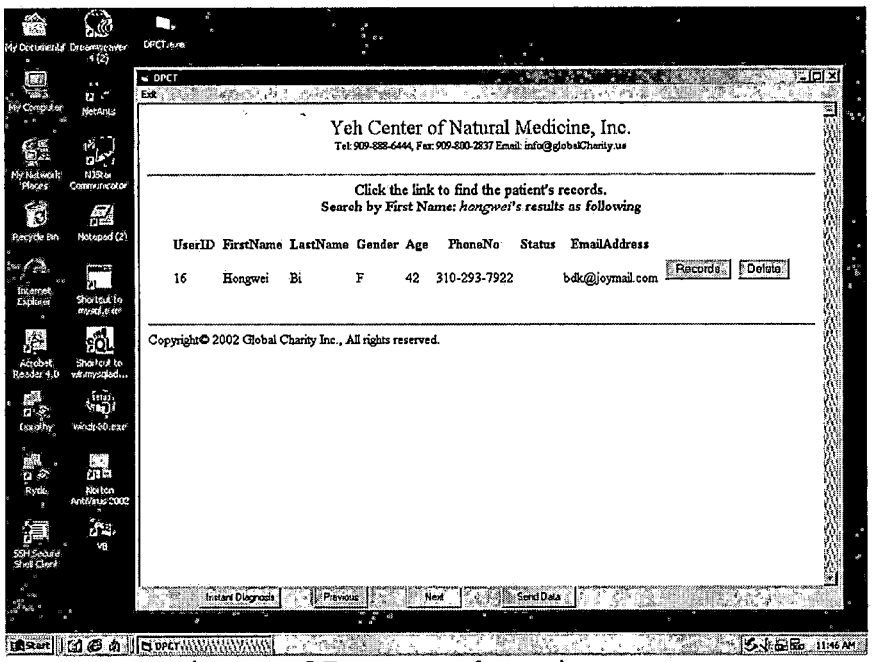


Figure 37. SearchByFirstName

- SearchResult Page. When the user clicks the EmailAdddress link as in Fig.37, the user's records will be displayed. All test data will be displayed, according to the user's specification in the checkboxes.

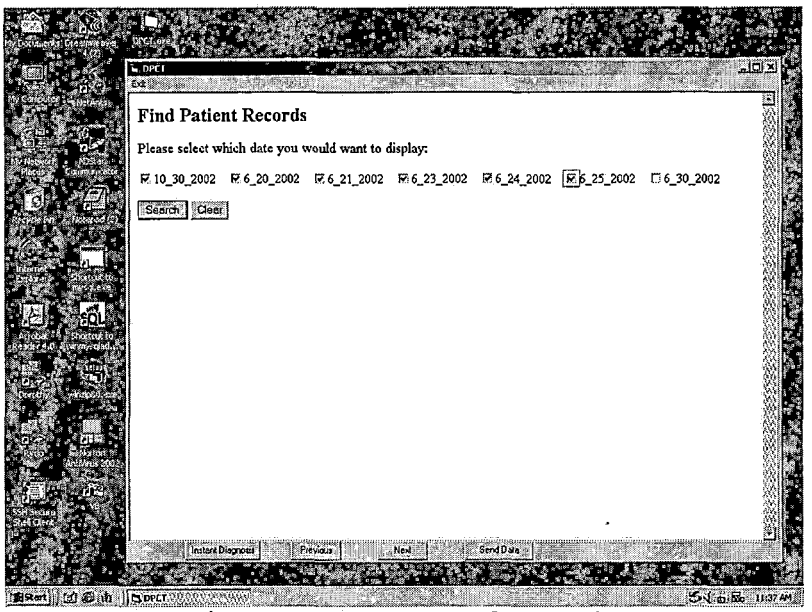


Figure 38. SearchResult

- SearchCaseHistory Page. This page displays the search case history results. We checked the first six check boxes in Fig.38. Then the screen in Fig.40 is displayed. Beside each diagnosis diagram there is a date link. This link connects to the specified diagnosis item. The interface is the same as Diagnosis.asp Figure 35.This page is designed for comparing different periods.

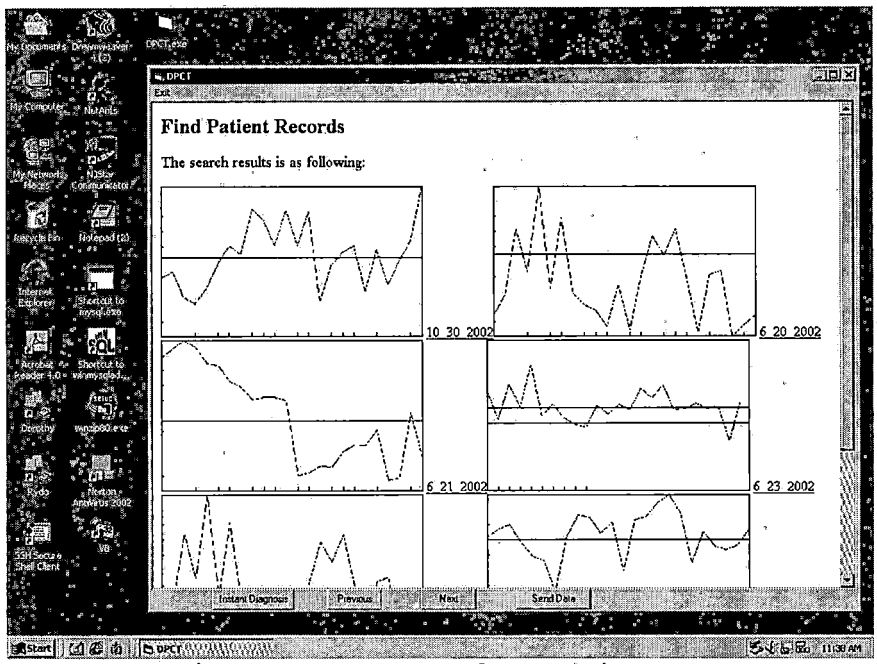


Figure 39. SearchCaseHistory

- SearchByLName Page. This page displays the search result by searching the Last Name is "Bi".

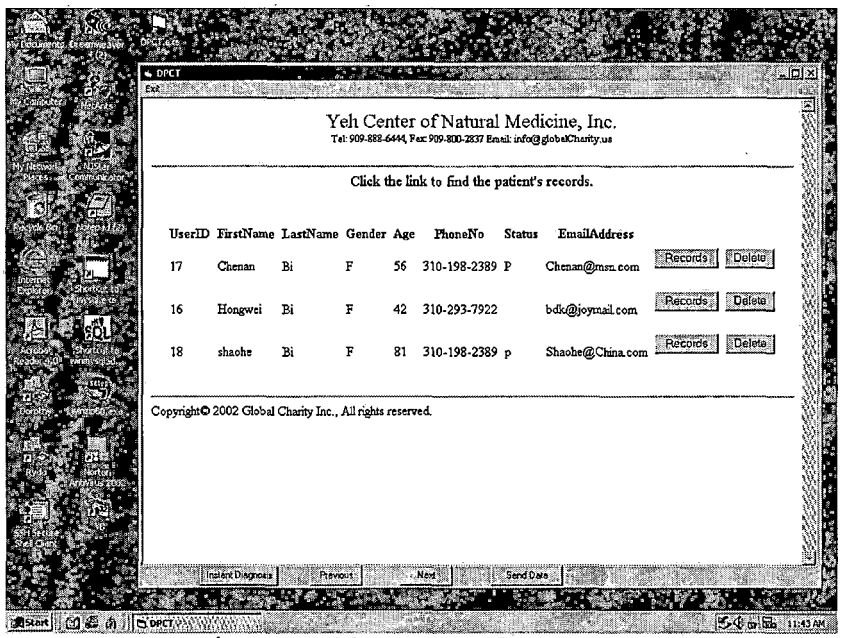


Figure 40. SearchByLName

- SearchByGender Page. This page displays the search by gender result.

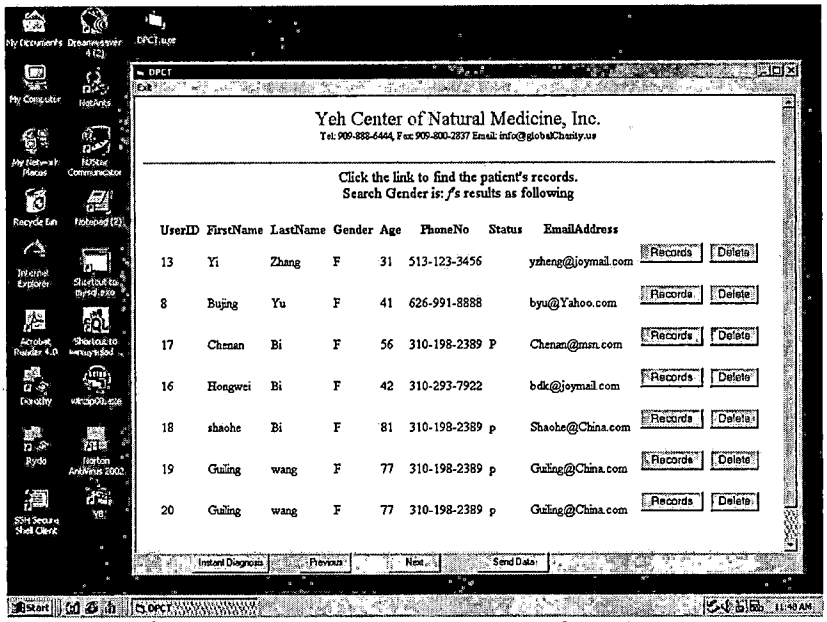


Figure 41. SearchByGender

- Add New User Page. This page is used by the physician to add a new user. The page is called by Sub_Homepage (Figure 34)

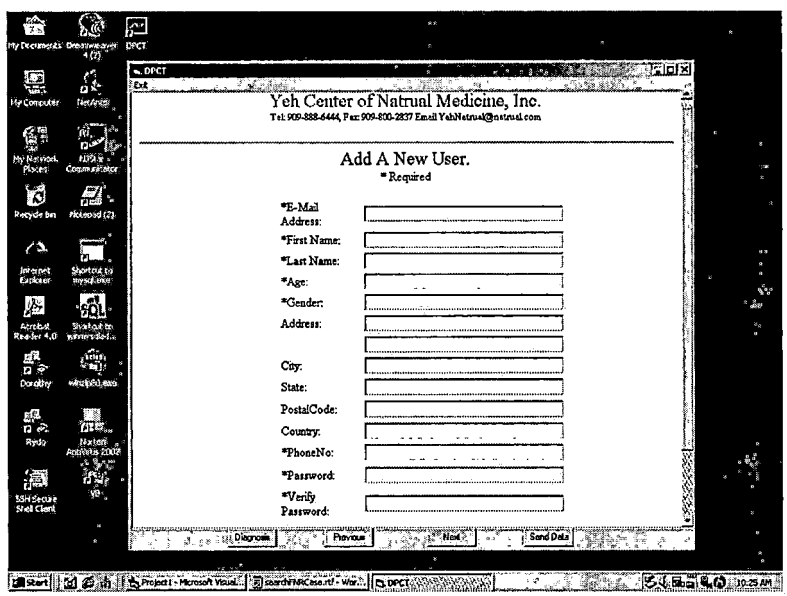


Figure 42. Add New User

Functions

Whenever a user wants to use this product, the user ID and password are required. Users can escape this system just click "Exit" menu bar or close the interface window. Only those users who have the DPCT account can access DPCT Web system. The system shall allow physician checking related patient records and giving instant diagnosis through typing the related patient's information. Patients need input their symptoms and 24 source point data to Web server and notice their physician for getting instant diagnosis. User can click the related date to get detailed diagnosis information.

For detail sequence of operations refer Figure 49. and Figure 50.

Schema Diagram:

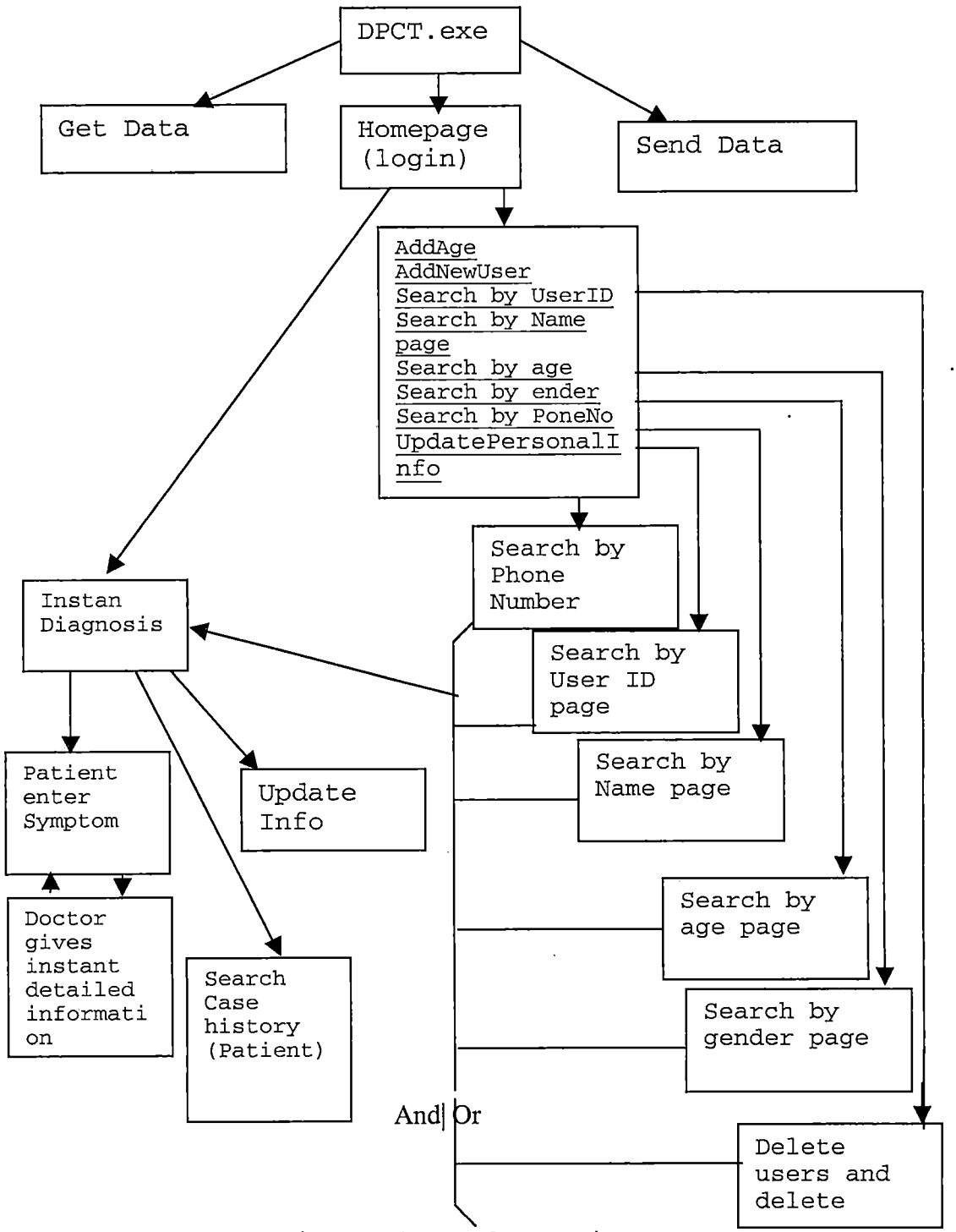


Figure 43. Schema Diagram

Performance Requirements

This product supports all terminal users under traffic allowing. All users are designed under session mode. ASP tracks every user so there is no mutual exclusion considered in the software system.

It takes 10 minutes to traverse the 24 source points for body's detection.

Logical Database Requirements.

For details refer to Chapter Four Data Base Tier.

Design Constraints

This system hardware interface with connection to computer is restrained in serial port.

Standards Compliance

The data structure and algorithms will comply with those accepted in publicly available documents and texts.

Software System Attributes

- Reliability. The FTP server and ASP engine where DPCT will be located should be fictional.
- Availability. For server side, any W32 with 64M RAM equipped and ASP and FTP server installed should available. For client side, any PC with World Wide Web access can access DPCT locally.

- Security. Only those people who have account in the DPCT system can access DPCT Web system. Everyone should provide User identifier and password to access the system.
- Maintainability. Refer to chapter Six for details.
- Portability. Data gathering software is a standalone executable file that locates on patient local machine. It can install on any PC. The Web system software is placed on a server. The software system can be embedded in all windows OS. 4% components are created and registered in presentation tier and diagnosis tier. All Web system software is host dependent.

CHAPTER FOUR

DETAIL DESIGN

Hardware Design Architecture

DPCT's hardware design, Detection Device (DD), mainly has two parts: A/D converter and interface design. It is the hardware tier of the DPCT system. A 220V/5.5v/12V transformer is needed as power supply. The interface between DD and patient is a connector that gathers patient's body circuit values used as DD input values. Serial port COM1 is used as the interface between DD and PC. Through IRQ, the input analog values will be converted into digital binary numbers stored in the patient's hard disk. In order to reduce the distortion, between DD and PC serial port, a RS232 transceiver is used.

Power Requirements: V_{DD} +2.7 V/+5.5 V/+12V

I_{DD} 4 mA max

Analog Input: Voltage Range 0 ~ +V_{DD}
Input Current (+/-) 5μA max

Logic Input: Input High Voltage, (V_{INH}) V_{DD} = +2.0 V

Input Low voltage, (V_{IN1}) V_{DD} = +0.8 Vmax

Input Current: I_{IN} = (+/-)10 μA max

Digital Input: -0.3 V to +5.5 V

Conversion Rate: Conversion Time = 9 μs max

Track/Hold Acquisition Time 1.5 μs max

Temperature ranges: 0 ~ +70 C

Hardware Interface

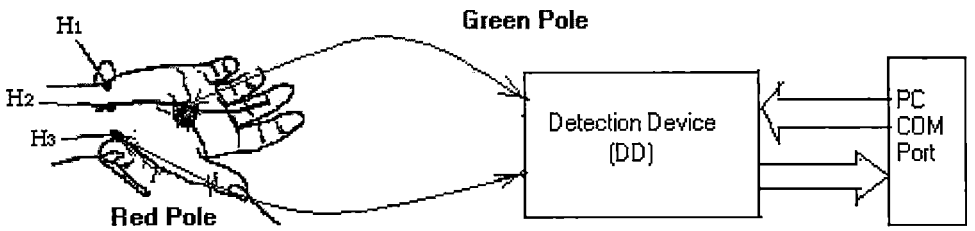
The interface between DD and patient is a connector that has two parts, the right part (red pole) and the left part (green pole). The patient places the red pole in the middle of the palm and then touches put the green pole to the 24 source points one by one.

The DD connects to the PC through COM1 port. Detected data will be stored in file in the patient's hard drive. Transmission procedure:

1. When computer is ready to accept data, it sends a signal (S1) to the DD.
2. When the DD receives the signal (S1), which comes from the COM1 port, it begins to convert the analog input into digital data. After conversion, the DD notices PC (S2) that it is ready also.
3. After the PC gets the signal (S2), the PC begins to send a clock pulse to stimulate the DD to get the converted data (Data).

4. When the PC receives the converted data, it stores them in file.

The physical diagram is as following:



Detection Device Interface

Figure 44. Detection Device Interface

Presentation Logic

The presentation logic forms the client tier. It generates the user interface between the patient and console. Also it is a standalone executable file that makes an Internet connection. It formats the data and uploads the data to the central server (IIS), through its connection to the Internet. Through the interface, the patient can only see what is presented by the presentation logic, so they will not be allowed to communicate with the data store directly. DPCT's central server dynamically generates all ASP pages.

The following are executable .DLL files and the ASP scripts that form the presentation logic:

Dpct.exe, serial.dll, default.asp, login.asp, updataPersonInfo.asp, changePassword.asp, searchByEmail, searchByFirstName.asp, searchByLastname.asp, searchByAge.asp, searchByGender.asp, searchByPhoneN.asp, searchBySympton

The DPCT application's icon is placed on the desktop. When user clicks the DPCT's icon (Figure 45.), the application will begin to run.

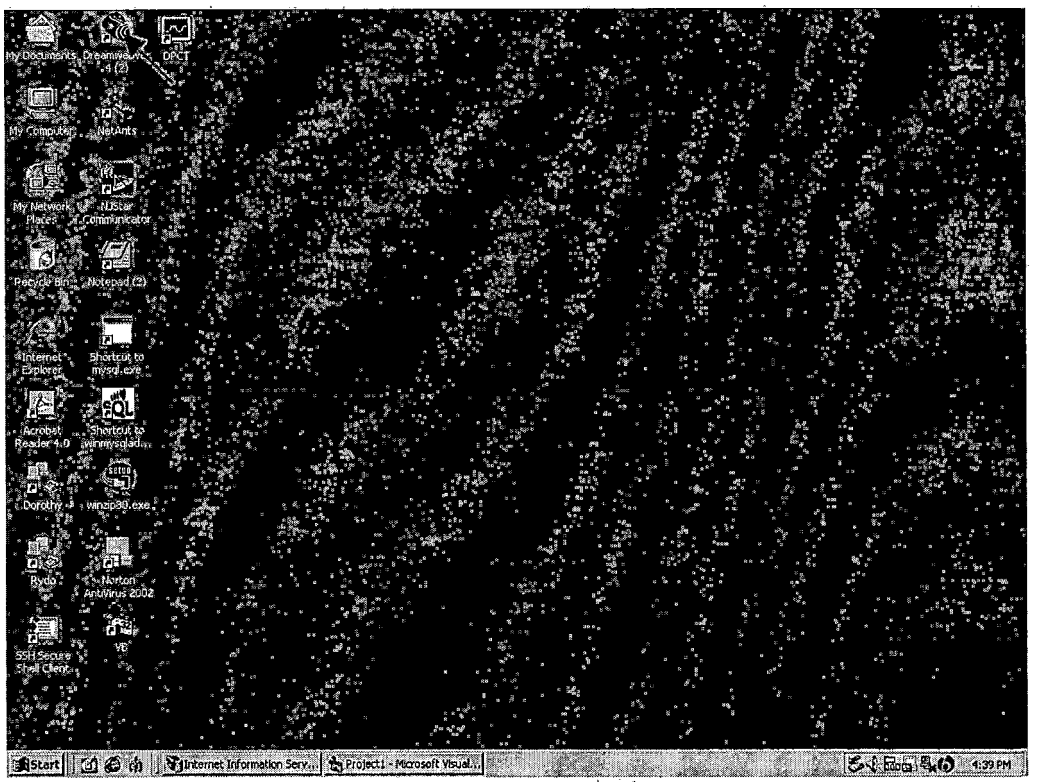


Figure 45. Excutable Icon

The first page that displays in front of the user is DPCT.exe's main form (Figure 5) embedding DPCT's default

Web page. For a Detailed interface design, please see section 4.2 the Graphical User Interface.

The presentation tier is also responsible for data transmission. The transmission is performed by a component Comftp. Once the Comftp is called, the data file will be sent to server side. Whenever a user wants to exit the application, just select the exit menu. The user can access the DPCT Web page through the embedded Web page, and following the instructions, the user can get the diagnosis result, and view the diagnosis history.

Diagnosis Tier Interface

Diagnosis tier consists of server side ASP scripts. These applications access a database. The interfaces of the diagnosis tier will be called by the presentation logic, and will retrieve all the information needed for presentation from the DPCT database.

Diagram of Diagnosis tier is as following:

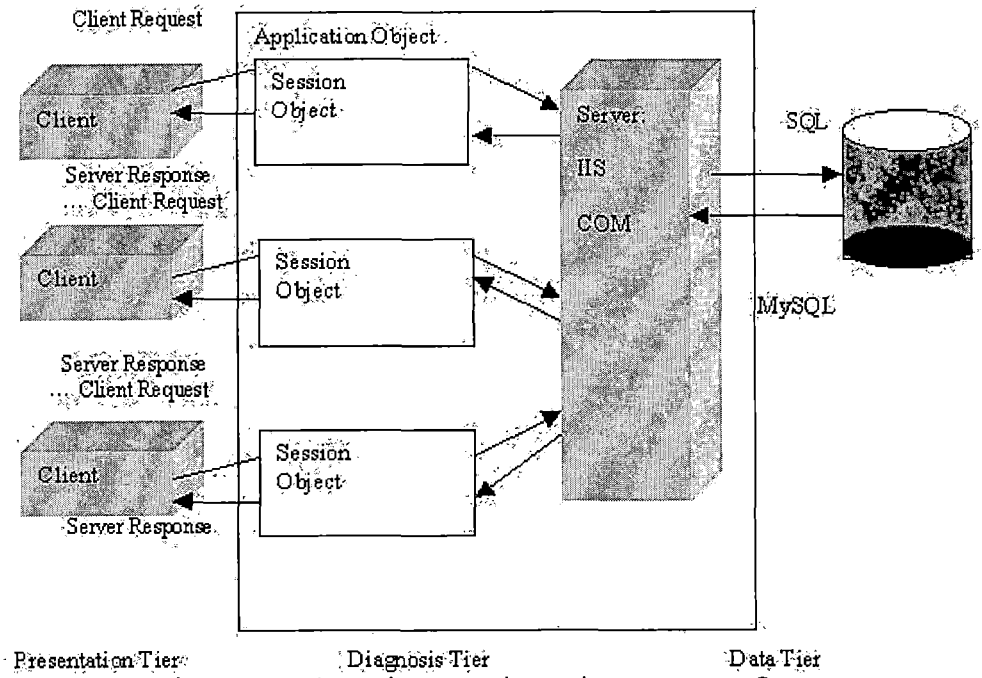


Figure 46. Diagnosis Tier Interface

Component Architecture

In order to increase the application's scalability and flexibility, COM-based programming was used as the core of the application. Both Presentation and Diagnosis tiers are designed as components.

In the Presentation tier, a component ComSerial was designed to gather diagnosis data.

In the Diagnosis tier, a component called ComUser was designed for communication between ASP and data store. By using ComUser, DPCT can easily add users, delete users, and connect to the server.

Diagram of the component architecture is as follows:

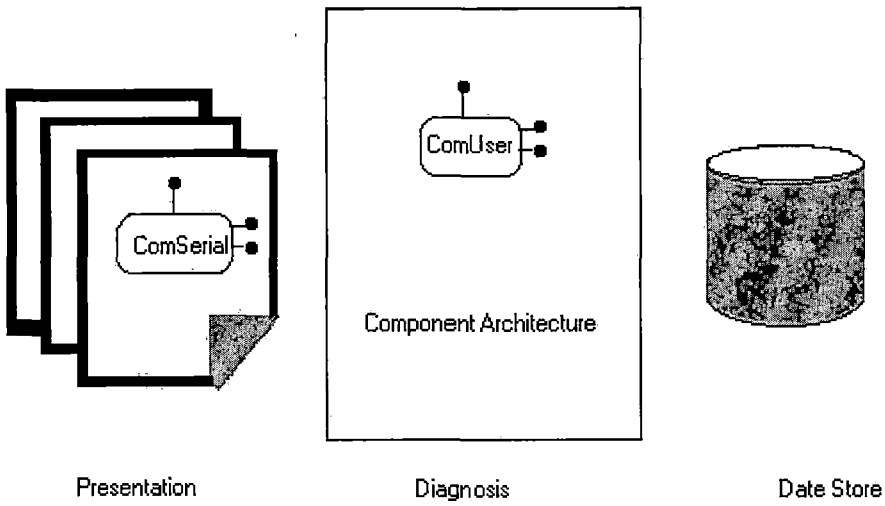


Figure 47. Component Architecture

Data Tier Design

DPCT's database is designed according to the relational model. DPCT utilizes MySQl as its database server. There are four tables to be designed in DPCT application. They are: User table, Diagnosis table, Points table, and Symptom table. These tables can be accessed by using ODBC connection in SQL through ADO, the best of the existing Microsoft data access programming models. The data stores can only be accessed by Diagnosis tier.

Database Design

- ER Diagram. DPCT database ER Diagram:

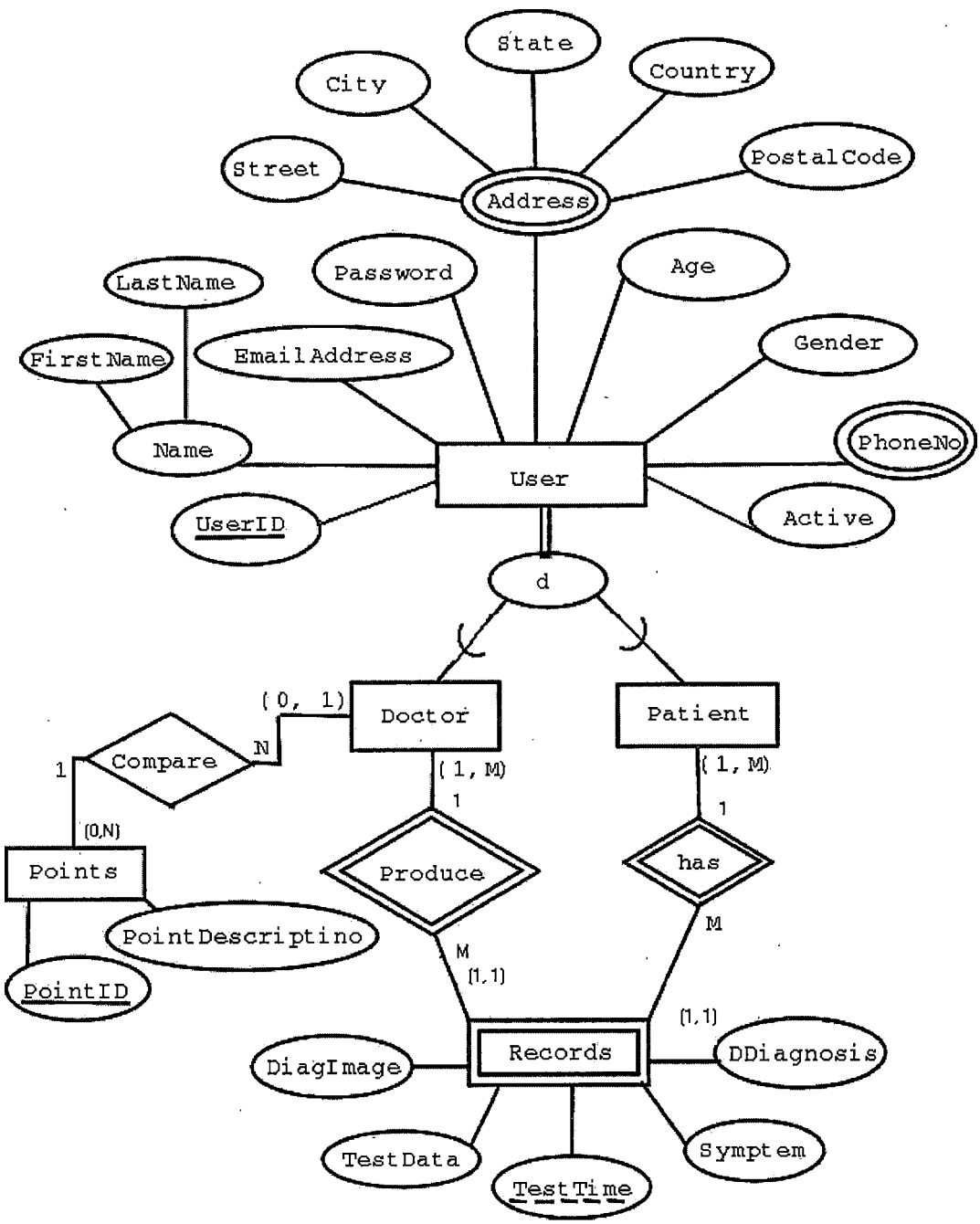


Figure 48. ER Diagram

- Data Dictionary. Table 2-5 shows different data tables.

Table 2. Records Table:

Field	Data Type	Description
UserID	BIGINT	Composite Key
DDiagnosis	Text	Doctor's diagnosis
Symptom	Text	Patient's symptom
TestTime	DateTime	Composite Key (Date and time the user login)
TestData	FLOAT	Diagnosis Data
DiagImage	BLOB	

Table 3. Address Table:

Field	Data Type	Description
Street	Text	Composite Key
City	Text	Composite Key
State	Text	Composite Key
Country	Text	Composite Key
PostalCode	Text	
UserID	BIGINT	Foreign Key

Table 4. User Table:

Field	Data Type	Description
UserID	BIGINT	Unique identifier
FirstName	Text	User's first name (required)
LastName	Text	User's last name (required)
EmailAddress	Text	User's e-mail address - used for login (required)
Password	Text	User-defined password - used for login (required)
Age	INT	
Gender	Char	
Active	Boolean	Flag indiction that the user is currently an active user
Status	Char	Indicate the current user is a patient or a doctor (required)

Table 5. Points Table:

Field	Data Type	Description
PointID	Char (2)	Unique identifier
PointDescription	Text	

Table 6. Phone Table:

Field	Data Type	Description
PhoneNo	VarChar(15)	Primary Key
UserID	BIGINT	Foreign Key

- SQL Commands

USE DB

SHOW Tables

DESC table

CREATE DATABASE db

CREATE TABLE table (column datatype [, column datatype]…)

Example:

Create table Points (PointID char(2), PointDescription Test);

DROP TABLE table

SELECT select_list From table [Where condition] [Order by column] [ASC | Desc]

Example:

Select * from User where LastName="Bi" group by TestTime;

SELECT select_list From table [AS alia_name] [, table [AS alia_name]] Where JoinCondition

INSERT [INTO] table VALUES (values_list)

DELETE FROM table [Where condition]

UPDATA table SET column = expr [, column = expr, …)

- Data Loading

1. Using INSERT [INTO] table VALUES (values_list)

2. Using LOADFILE (file_name)

UPDATA table_name SET column = LOADFILE(full pathname) where mcondition

Integration Architecture

The main methodology for the DPCT system design is a hybrid design. It contains four tiers: hardware tier, presentation tier, diagnosis tier, and database tier.

Presentation tier and DD directly control the hardware tier. After getting 24 data points, the presentation tier will send the data to the server for further manipulation.

Once the presentation tier sends data to diagnosis tier, it is almost finished its tasks. The Diagnosis tier is the glue between the database and user browser. It retrieves data from the database based on the client's requirement.

These Web pages need to be dynamic and have good integration to database. Refer to ASP 3.0's many attractive features, such as scriptless, new flow control capabilities, server scriptless, etc. It creates pages on

flying with good performance. So the entire Web pages were designed as ASP pages.

CHAPTER FIVE

TESTING

Testing is extremely important for hardware and software designs. The DPCT system testing includes two sections: Hardware and software testing.

Doctors and patients should follow the following sequential diagrams:

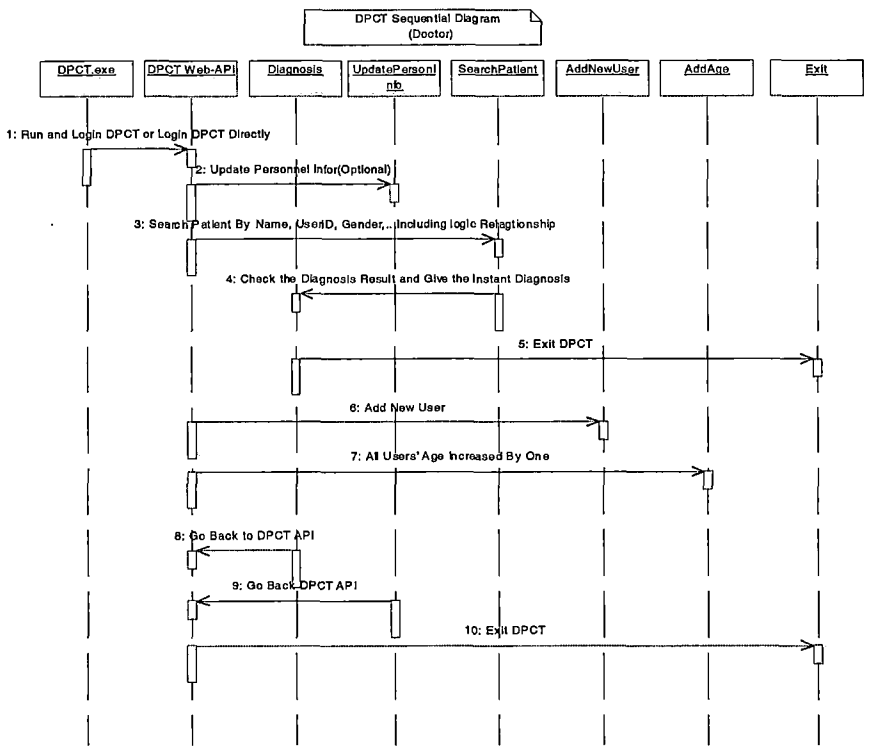


Figure 49. DoctorSequential Diagram

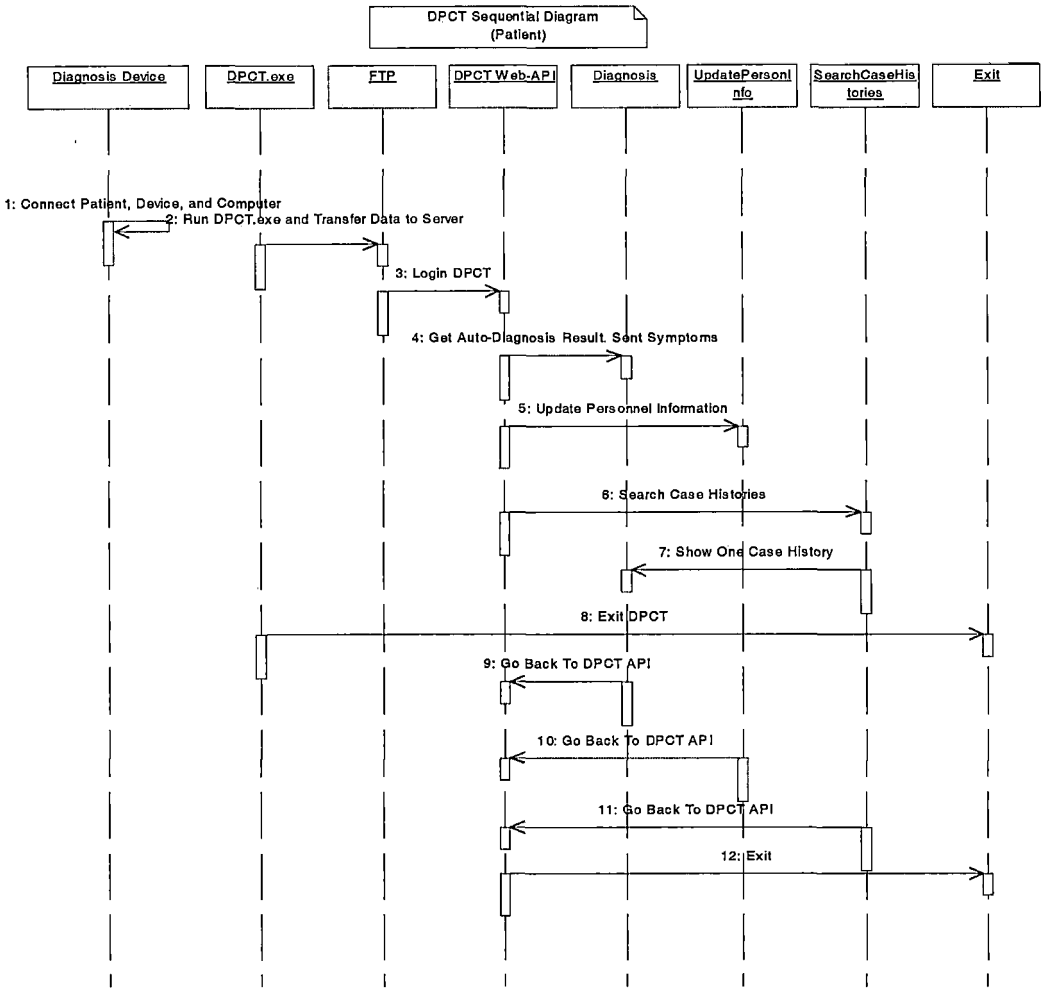


Figure 50. PatientSequential Diagram

Hardware Testing

Table 7. TestVCC1

Test Item	Desired Result	Actual Result
Input Voltage Vcc1 After filter.	5 V	4.998 V

Table 8. TestVCC2

Test Item	Desired Result	Actual Result
Input Voltage Vcc2 After filter.	13 V	12.99 V

Table 9. TestPointData

Test Number	Actual Result (mA)	Desired Result (mV)
1	8	8
2	8.66	8.33
3	7.86	8.173
4	8.246	8.1912
5	8.12	8.17

Mean=8.165

Dviation=$((8-8.165)^2+(8.66-8.165)^2+(7.86-8.165)^2+(8.246-8.165)^2 + (8.12-8.165)^2)^{1/2} / 5 = 0.122$

Software Testing

Table 10. TestCheckLogin.asp

Test Item	Desired Result	Actual Result
CheckLogin.asp page	Login the user account if the user has an account in DPCT. The wrong Email address and password will request user input again.	The same as Desired.

Table 11. TestAddNewUser.asp

Test Item	Desired Result	Actual Result
AddNewUser. asp page	After add a new user, Doctor can search the patient information.	The same as Desired.

Table 12. TestAddAge.asp

Test Item	Desired Result	Actual Result
AddAge.asp page	After click the Add Age button, all patients age increase by one.	The same as Desired.

Table 13. TestUpdateInfo.asp

Test Item	Desired Result	Actual Result
UpdateInfo. asp page	After update user's information, all the user's record updated.	The same as Desired.

Table 14. TestSearchCaseHistory.asp

Test Item	Desired Result	Actual Result
SearchCaseHistory.asp page	display all the patient record lists.	The same as Desired.

Table 15. TestSearchByFirstName

Test Item	Desired Result	Actual Result
Search.asp page	Display all patients whose first names are the same as required.	The same as Desired.

Table 16. TestSearchByLastName

Test Item	Desired Result	Actual Result
Search.asp page	Display all patients whose first names are the same as required.	The same as Desired.

Table 17. TestSearchByAge

Test Item	Desired Result	Actual Result
Search.asp page	Input the age. Display all patients whose ages are the same as required.	The same as Desired.

Table 18. TestSearchByGender

Test Item	Desired Result	Actual Result
Search.asp page	Input "F" or "M". Display all female or male patient record	The same as Desired.

Table 19. TestSearchByUserID

Test Item	Desired Result	Actual Result
Search.asp page	Input the patient UserID. Display the patient record.	The same as Desired.

Table 20. TestSearchByPhoneNo

Test Item	Desired Result	Actual Result
Search.asp page	Input the patient's phone No. Display the patient's record.	The same as Desired.

Table 21. TestSearchByANDLogic

Test Item	Desired Result	Actual Result
Search.asp page	Display all patients who match the logic AND relationship.	The same as Desired.

Table 22. TestSearchByORLogic

Test Item	Desired Result	Actual Result
Search.asp page	Display all patients who match the OR condition.	The same as Desired.

CHAPTER SIX

MAINTENANCE MANUAL

Files and Directories

DPCT system has three main directories. One is at E:\Inetpub\WWWROOT\Ryodoraku\, E is the drive that installs IIS server. All ASP pages are stored here. One is at E:\Inetpub\FTPROOT\, all patients' diagnosis data and images are stored in their own directories. Each patient has his own directories named by their UserID. The third main directory is at E:\DPCT\ used to store DPCT.vbp. The detailed distribution is as below:

```
E:
|----Inetpub
      |----WWWROOT
            |----Ryodoraku
                  |----Default.asp
                  |----Login.asp
                  |----Datastore.asp
                  |----AddNewUser.asp
                  |----AddNewUserConfirm.asp
                  |----InstantDiagnosisP.asp
                  |----InstantDiagnosisD.asp
                  |----UpdateInfo.asp
```

```
                    |----checkLogin.asp
                    |----Search.asp
                    |----CurTime.asp
                    |----StrToTable.asp
                    |----DoctorConfirmation.asp
                    |----PatientConfirmation.asp
                    |----DeleteDiagnosis.asp
                    |----DeleteDiagConfirm.asp
                    |----DeleteUser.asp
                    |----AddNewUser.asp
                    |----AddAge.asp
                    |----PatientSubPage.asp
                    |----SearchCaseHistory.asp
                    |----Footer.asp
                    |----Head1.asp
                    |----Head2.asp
E:
|----Inetpub
       |----FTPROOT
             |----1
                   |----TestDate.txt
                   |----TestDate.bmp
             |----2
                   |----TestDate.txt
```

```
                |----TestDate.bmp
          |----3
                |----TestDate.txt
                |----TestDate.bmp
             . . .
          |----LastPatient
                |----TestDate.txt
                |----TestDate.bmp
E:
|----DPCT
      |----DPCT.vbp
      |----DPCT.exe
```

Software Installation

The DPCT system was developed under Windows 2000 accompanied with Microsoft IIS and FTP server and SMTP. What we need to do is to download and install MySQL server. We can download it from: http://www.mysql.com/downloads/mysql-3.23.html. The detailed installation procedure is as below:

1. Copy DPCT.exe to local machine. Create a DPCT directory under local drive is recommended.
2. Copy RYODORAKU directory to E:\Inetpub\WWWROOT\Ryodoraku

3. Create user Directories under

   E:\Inetpub\WWWROOT\.

   The directories names are named by Users' UserID.

   Whenever add a new user, DPCT database will

   create a new record in User table accompanied

   with an unique UserID automatically.

   Administrator or doctors can find the UserID

   through DPCT search funtionaly.

4. Start IIS and FTP server.

Database Administration

After install MySQL server on the local machine, Administrator can create the tables and input the automatic diagnosis data into Points table. Also the administrator can utilize Mysqldump to load the tables and data.

For example:

Mysql>mysqldump -C -h domain -u user -pPassword Database > *.sql

Mysql>mysqldump -C -h domain -u user -pPassword Database > D:\localpath.sql

The administrator can setup new users and grant their privilege.

DPCT system was designed more flexible for normal user. Doctors can update, delete, and insert records into database through DPCT system ASP pages.

CHAPTER SEVEN

CONCLUSION AND FUTURE DIRECTIONS

Conclusion

High-speed, high-reliability, high quality, and convenience have become the core specifications of modern products. Due to the huge pressure of our daily life, our bodies need to be cared more; we need to know our bodies' situation more. But going to clinic to get instant diagnosis frequently is inconvenient for patients. Recently, Internet has been applied to many fields, resulting in increased efficiency. DPCT utilizes the Internet in the same way to achieve increased efficiency for patients and doctors. People can get diagnosed anywhere by login in the Internet. Also, people can diagnose themselves as many times as they want. This provides the possibility of detecting disease in our bodies anytime.

From the detailed hardware design and assembling, software plan and coding, Web site design and building, database design and constructing, I accomplished the whole DPCT system. In order to meet its requirement specification, IIS, FTP server, Visual Basic, VBScript and JavaScript, and ASP were used with. Auto-diagnosis and

data gathering were created by components coded in Visual Basic. So this system supposes to achieve good integration, flexibility, scalability, and compatibility.

By adding a database into this project, DPCT is also easy to be managed by patient and doctor. The doctor can add or delete patient records and items anytime. Patients can update their personal information and view their case histories without time and place restraints.

Besides benefit both physicians and patients. This project also benefits TCM practitioners and medical professionals for case study. It shows its own outstanding convenience when patients are on their traveling. They can communicate with their physician instantly and conveniently. This system can be used as an aid tool for serious and dangerous conditions detection and delivering. It increases the accuracy and instance when people are in the urgent notification. All the instant diagnosis need to incorporate doctor notification through email or telephone call.

Future Directions

1. This project can be developed by Chili!ASP, or Halcyon Sofrtware's Instant ASP, even JSP to enlarge its scalabilities. Chili!ASP utilizes the

same development tools and functionality as ASP, but it suits more environments, such as Netscape, Lotus Go, as well as NT 4.0-based Web servers. Halcyon Software's Instant ASP has many development platforms; it runs on a whole range of Web servers, application servers, and operating system platforms. These are: Windows NT, Linux, Apache, Sun, Novell, AIX, AS/400, S/390, Netscape, Websphere, etc. The other approach is to change the ASP code to JSP, which runs on both Windows and Unix platforms.

2. Audio and video functionalities can be included to increase DPCT's attractiveness and reactivity.
3. In order to access most data stores, OLE-DB, an underlying technology, is recommended for data connectivity. OLE-DB can connect any kind of database. It provides large scalability.
4. DPCT can be improved by a neural network (NN) or a neural network simulator (NNS), that learns from collected case histories. Using NN or NNS, the software becomes more flexible and compatible, and can be used in other projects that are similar to this environment, hence increasing the project's usability. It may be

based on fuzzy logic and NN combinations written in Visual C++.

5. In order to benefit physician's notification, an auto-waked system should be added in physician side. Whenever a patient sends a request for instant diagnosis, an indicator should show up in the physician system interface.

APPENDIX A

SOURCE CODE

```
'*********************************************************************'
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: checkLogin.asp
'Programmer: Hongwei Bi
'Date: 04/08/02
'Description: Users must enter their E-mail and password 'correctly
to login the DPCT system
'*********************************************************************'
<!--#include file="Datastore.asp"-->
<%
        Dim strName, strValue
        Dim strEmail, strPassword
        strEmail = Request("Email")
        strPassword = Request("Password")
        Dim rsUsers
        set objConn = Server.CreateObject("ADODB.Connection")
        set rsUsers = Server.CreateObject("ADODB.Recordset")
        strSQL = "SELECT * FROM User WHERE EmailAddress = '" &
strEmail & "';"
        objConn.Open strConnect
        rsUsers.Open strSQL, objConn

        If rsUsers.EOF then
          Session("EmailAddress") = Request("Email")
           if Request("SecondTry") = "True" then
                 Response.Redirect "login.asp?NotFound=True"
           Else
       ' User not found
       ' User's had two goes
                 Response.Redirect "login.asp?SecondTry=True"

       ' - must reLogin
           end  if
        Else
       ' Username wrong; password wrong
       ' - allow another go
       'One or more users found - check password
          While Not rsUsers.EOF
```

```
        If UCase(rsUsers("Password")) = UCase(strPassword) Then
    ' password matched
          For Each strField in rsUsers.Fields
          strName = strField.Name
   ' populate session variables
          strValue = strField.value
          Session(strName) = strValue
          Next
     Session("blnValidUser") = True
     Response.Redirect "SubHomepage.asp"
   ' successful login
       Else
         rsUsers.MoveNext
       End If
     Wend
     Session("EmailAddress") = Request("Email")
' if we get this far then...
' ...password doesn't match any of DB entries
    If Request("SecondTry") = "True" then
' Users have to relogin
    Response.Redirect "login.asp?SecondTry=True&WrongPW=True"
     End If
 end if
%>
```

```
'*********************************************************************'
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: PatientConfirmation.asp
'Programmer: Hongwei Bi
'Date: 04/08/02
'Description: After the patient update his information,
'confirmation page prompts out to confirm the entered info.
'*********************************************************************'
<!-- #include file = "Datastore.asp" -->
<%
      Dim rsUsers, objConn, loginStatus, ID
      set objConn = Server.CreateObject("ADODB.Connection")
      set rsUsers = Server.CreateObject("ADODB.Recordset")
      objConn.Open strConnect
      ID = session("userID")
      rsUsers.Open "select * from User where userID = " & ID,
objConn, adKeyset, adLockOptimistic, adCmdText
%>
<html>
<head>
<title>Confirmation Page</title>
</head>
<body>
<center>
<b><font size="+2" color="#004000">
      Doctor-Patient Communication Tool (DPCT)</font>
<font size="2">
      Tel: 909-888-6444, Fax: 909-800-2837 Email:
info@GlobalCharity.us</font>
</b>
<hr><p><font size=+1 color = darkgreen>User: <i>  <%=
rsUsers("firstName")& " "& rsUsers("LastName") %></i>'s information
has been updated.</font>
</center>
<blockquote><blockquote><blockquote><blockquote><blockquote>
<center>
   <table  width = "400">
      <tr>
```

```
      <td align = left width ="50%">
      FirstName:  </td>
      <td align =left width = "50%">
      <%= rsUsers("FirstName")%> </td></tr>
<tr>
      <td> LastName: </td>
      <td align = left width ="50%">
      <%= rsUsers("LastName")%>  </td></tr>
<tr>
      <td align = left width ="50%">
      Age:  </td>
      <td align = left width ="50%">
      <%= rsUsers("Age")%>    </td></tr>
<tr>
      <td align = left width ="50%">
      Gender: </td>
      <td align = left width ="50%">
      <%= rsUsers("Gender")%> </td></tr>
<tr>
      <td align = left width ="50%">
      Address: </td>
      <td align = "left" width ="50%">
      <%= rsUsers("StreetAddress1")%> </td></tr>
<tr>
      <td align = left width ="50%">
      City: </td>
      <td align = "left" width ="50%">
      <%= rsUsers("City")%> </td></tr>
<tr>
      <td align = left width ="50%">
      State: </td>
      <td align = "left" width ="50%">
      <%= rsUsers("State")%> </td></tr>
<tr>
      <td align = left width ="50%">
      PostalCode: </td>
      <td align = "left" width ="50%">
      <%= rsUsers("PostalCode")%> </td></tr>
```

```
      <tr>
            <td align = left width ="50%">
            Country: </td>
            <td align = "left" width ="50%">
            <%= rsUsers("Country")%> </td></tr>
      <tr>
            <td align = left width ="50%">
            PhoneNo: </td>
            <td align = "left" width ="50%">
            <%= rsUsers("PhoneNo")%>   </td></tr>
      <tr>
            <td align = left width ="50%">
            Password: </td>
            <td align = "left" width ="50%">
            <%= rsUsers("Password")%> </td></tr>
</table><p> <font size="4">
         <a href="diagnosisDP.asp">Diagnosis</a>    
         <a href = "subHomepage.asp">Search Case History </a>
           </font></p>
</center>
</blockquote></blockquote></blockquote></blockquote></blockquote>
<!-- #include file = "note.asp" -->
</body>
</html>
```

```
'*********************************************************************'
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: DoctorConfirmation.asp
'Programmer: Hongwei Bi
'Date: 04/08/02
'Description: After the doctor update his information, 'confirmation
page prompts out to confirm the entered 'info. The difference between
"DoctorConfirmation.asp" and '"PatientConfirmationP.asp" is the
former one for the 'doctor which has a button connected to
"addNewUser.asp" 'page and a button to update users ages.
'*********************************************************************'
      <!-- #include file = "Datastore.asp" -->
      <%
      Dim rsUsers, objConn, loginStatus, ID
         set objConn = Server.CreateObject("ADODB.Connection")
         set rsUsers = Server.CreateObject("ADODB.Recordset")
         objConn.Open strConnect
          ID = session("userID")
         rsUsers.Open "select * from User where userID = " & ID,
objConn, adKeyset, adLockOptimistic, adCmdText
      %>
      <html>
      <head>
         <meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
         <meta http-equiv="Content-Language" content="en-us">
         <meta name="GENERATOR" content="Microsoft FrontPage 4.0">
         <meta name="ProgId" content="FrontPage.Editor.Document">
         <title>Confirmation Page</title>
      </head>
      <body>

      <center>
        <b><font size="+2" color="#004000">Doctor-Patient
        Communication Tool (DPCT)</font>
        <font size="2">Tel: 909-888-6444, Fax: 909-800-2837 Email
info@GlobalCharity.us</font>
        </b>
```

```
        <hr>
        <p><font size=+1 color = darkgreen>User: <i>  <%=
rsUsers("firstName")& " "& rsUsers("LastName") %></i>'s information
has been updated.</font>
      </center>
      <blockquote><blockquote><blockquote><blockquote>
         <center>
         <table  width = "400">
            <tr>
               <td align = left width ="50%">
               FirstName:  </td>
               <td align =left width = "50%">
               <%= rsUsers("FirstName")%> </td></tr>
           <tr>
               <td>
              LastName: </td>
               <td align = left width ="50%">
              <%= rsUsers("LastName")%>  </td></tr>
           <tr>
               <td align = left width ="50%">
               Age:  </td>
               <td align = left width ="50%">
               <%= rsUsers("Age")%>    </td></tr>
           <tr>
               <td align = left width ="50%">
              Gender: </td>
               <td align = left width ="50%">
              <%= rsUsers("Gender")%> </td></tr>
          <tr>
             <td align = left width ="50%">
             Address: </td>
              <td align = "left" width ="50%">
              <%= rsUsers("StreetAddress1")%> </td></tr>
          <tr>
               <td align = left width ="50%">
               City: </td>
              <td align = "left" width ="50%">
               <%= rsUsers("City")%> </td></tr>
```

```
    <tr>
         <td align = left width ="50%">
         State: </td>
         <td align = "left" width ="50%">
         <%= rsUsers("State")%> </td></tr>
    <tr>
         <td align = left width ="50%">
         PostalCode: </td>
         <td align = "left" width ="50%">
         <%= rsUsers("PostalCode")%> </td></tr>
    <tr>
         <td align = left width ="50%">
         Country: </td>
         <td align = "left" width ="50%">
         <%= rsUsers("Country")%> </td></tr>
    <tr>
         <td align = left width ="50%">
         PhoneNo: </td>
         <td align = "left" width ="50%">
        <%= rsUsers("PhoneNo")%>    </td></tr>
    <tr>
         <td align = left width ="50%">
         Password: </td>
         <td align = "left" width ="50%">
         <%= rsUsers("Password")%> </td></tr>
   </table>
<p> <font size="4">
   <a href="diagnosisDP.asp">Diagnosis</a>    
   <a href = "PreAddNewUser.asp">AddNewUser.asp</a>
   <a href = "subHomepage.asp">Search Case History </a>
    </font></p>
</center>
</blockquote></blockquote></blockquote></blockquote>
<!-- #include file = "note.asp" -->
</body>
</html>
```

```
'*****************************************************************'
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: curTime.asp
'Programmer: Hongwei Bi
'Date: 04/08/02
'Description: Display the current time
'*****************************************************'
      <% curTime = now  %>
      <html>
      <head>
      <title>Display Current Time </title>
      </head>
      <body>
      <p><center> The current time is:
      <table>
         <tr>
            <td bgcolor="lightpink" width="50">
            <%= datePart("m",Now) & datePart("d", now) &
datePart("yyyy", now) %>
            </td></tr><tr>
      <%= year(date)&
month(date)&day(date)&hour(time)&minute(time)&second(time) %>
        </tr> </table>
      </center>
      </body>
      </html>
```

```
'*********************************************************************'
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: dataStore.asp
'Programmer: Hongwei Bi
'Date: 04/08/02
'Description: It is a SSI file that connects the ASP pages 'to
Database
'*********************************************************************'
      <!-- metadata type = "typelib"
            File ="D:\Program Files\Common
Files\System\ado\msado15.dll" -->

       <%
         strConnect = "driver={MYSQL}; SERVER=charity-3s7patc;
database=PROJECT; UID=sa; PWD =xxx"
      %>
```

```
'*****************************************************************'
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: default.asp
'Programmer: Hongwei Bi
'Date: 04/08/02'Description: This is the first page of the 'DPCT
system. 'From here user can retrieve Web page and 'access DPCT's
Database.
'*****************************************************'
      <html>
      <head>
      <title>DPCT Default.asp</title>
      <meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
      </head>
      <body bgcolor="FFFFFFe" text="#000000">
      <!-- #include file = "head1.asp" -->
      <div align="center">
         <font size="5" color="#004040"><br>
            Welcom to Doctor-Patient Communication Tool (DPCT)
         </font>
         <font size="5" color="#004040"><br>
            -- <font size="4">RYODORAKU Application on the Web
               </font>
         </font>
      </div>
      <p><font size="4">The Doctor-Patient Communiction Tool (DPCT)
is a Web-based application system. Patients can get diagnosed on the
Web. Doctor can communicate with the patient and give the instant
diagnosis throught the DPCT system on the Web.</font>
      <p><font size="4"><br>  <br>
        Please <a href = "login.asp">Click Here</a> to login the
system. </font></p>
        <p> </p>
        <p> </p>
        <p> </p>
      <!-- #include file = "note.asp" -->
      </body>
      </html>
```

```
'*********************************************************************'
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: deleteConfirm.asp
'Programmer: Hongwei Bi
'Date: 04/08/02
'Description: Once the patient's record has been deleted, 'this page
shows up to confirm the deletion.
'*********************************************************************'
      <!-- #include File = "Datastore.asp" -->
      <HTML>
      <HEAD>
      <TITLE>Delete Confirmation Patient Record</TITLE>
      <STYLE TYPE="text/css">
        BODY {font-family:Verdana,Tahoma,Arial,sans-serif; font-
size:10pt}
        TD {font-family:Verdana,Tahoma,Arial,sans-serif; font-
size:10pt}
      </STYLE>
      </head>
      <BODY BGCOLOR="#eefffF">
      <font size = "+2" face = "Times new Roman" >Delete Patient
Record</font>
      <HR>
      <br>
      <%
       Dim rsUsers, objConn , strT, ID
       set objConn = Server.CreateObject("ADODB.Connection")
       set rsUsers = Server.CreateObject("ADODB.Recordset")
       ID = CStr (Request.QueryString("Id"))
       objConn.Open strConnect
       If Session("Status") = "d" or Session ("Status") = "D" then
        objConn.Execute "Delete from User where UserID =" & ID, ,
adCmdText + adExecuteNoRecords
       objConn.Execute "Delete from Symptom where UserID =" & ID, ,
adCmdText + adExecuteNoRecords
       objConn.Execute "Delete from Diagnosis where UserID =" & ID, ,
adCmdText + adExecuteNoRecords
```

```
 Response.Write "Patient record deleted successfully."
       strT = "<Form name=deleteForm2 method = post action
=subHomepage.asp" & ">"
       strT = strT & "<input type = submit value=BackToHomePage Name
= B1>"
       strT = strT & "</Form>"
       Response.Write strT
       else
       Response.Write "Privilege Deny!"
       strT = "<Form name=deleteForm2 method = post action
=subHomepage.asp" & ">"
       strT = strT & "<input type = submit value=BackToSubHomePage
Name = B1>"
       strT = strT & "</Form>"
       Response.Write strT
       end if
      %>
      </BODY>
      </HTML>
```

```
'*********************************************************************'
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: DeleteDiagnosis.asp
'Programmer: Hongwei Bi
'Date: 04/08/02
'Description: When the doctor wants to delete the 'patient's 'record,
this page shows up to ask doctor 'whether he/she 'make sure to delete
the record. Once the 'patient's record 'has been deleted, it can't be
recovered.
'*********************************************************************'
      <!-- #include file = "DataStore.asp" -->
      <html>
      <head>
      <title>Delete Diagnosis</title>
      </head>
      <body>
      <p align="center"><font face="Times New Roman" size="5"><font
color="#006600">Doctor-Patient Communication Tool
      (DPCT)</font><span style="background-color: #0000FF"><br>
      </span></font>
      </p>
      <%
         Dim rsUsers, rsUsers2, objConn, DID, DrecordID
         set objConn = Server.CreateObject("ADODB.Connection")
         set rsUsers = Server.CreateObject("ADODB.Recordset")
         DID = Request.QueryString("Id")
         DrecordID = Request.QueryString("deleteID")
         objConn.Open strConnect
         rsUsers.Open "select * from user where userID = " & DID,
objConn, adOpenForwardOnly, adLockOptimistic, adCmdText
      curTime = now %>
      <table align = "right">
       <tr>
         <td  width="198">
          <%= curTime %>
         </td></tr>
      </table> <br>
      <center>  <blockquote>
```

```
      <table border="1" width="89%" bgcolor = "#EEffff">
        <tr>
          <td width="12%">Name </td>
          <td width="28%"><%=rsUsers("FirstName") & " " &
rsUsers("LastName") %></td>
          <td width="10%">Age</td>
          <td width="10%"><%= rsUsers("Age") %></td>
          <td width="5%">Gender</td>
          <td width="5%"><%= rsUsers("Gender") %></td>
          <td width="5%">Tel:</td>
          <td width="25%"><%= rsUsers("PhoneNo") %></td>
        </tr>
        <tr>
          <td width="12%">Address</td>
          <td width="88%" colspan="7"><%= rsUsers("StreetAddress1") &
" "& rsUsers("City") &", "&rsUsers("state")&" "& rsUsers("Country")
%></td>
        </tr>
      </table> </blockquote></center>
      <blockquote><blockquote><blockquote>
      <center>
      <img src = "D://inetpub/ftproot/<%= DID %>/<%= DrecordID
%>.bmp"  width= "600" height="300">
      </blockquote></blockquote></blockquote>
       <font color = brown size =4>Are you sure you want to delete
this diagnosis record?</font><BR><BR>
      <table>
         <tr>
            <td> Diagnosis Date: <%= DrecordID %> </td>
            <td> <form name = "communication" method = "post"  action
= "deleteDiagnosisConfirm.asp?Id=<%= DID %>&recordID=<%= DrecordID%>"
>
           <input type="Submit" value="Yes" name="SubmitDelete">
   
           <input type = "hidden" name= "DUserID" value = ID>
           <input type = "hidden" name = "recordID" value =
DrecordID>
                </form> </td>
```

```
            <td>
            <form name = "communication2" method = "post"  action =
"SubHomepage.asp">
            <input type="submit" value="No" name="ResetDiagnosis">
           </form> </td></tr></table> </center>
      <% rsUsers.Close
           set rsUsers  = nothing
      %>
      <p><a href="subHomepage.asp">BackHomepage</a></p>
      <hr>
      <p>Copyright©2002 Charity Globe Inc. All right reserved.</p>
      </body>
      </html>
```

```
'*********************************************************************
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: diagnosis.asp
'Programmer: Hongwei Bi
'Date: 04/08/02
'Description: This page is displayed in the doctor side. 'displays the
automated diagnosis results and patient's 'symptom. The doctor also can
input the detailed 'diagnosis.
'*********************************************************************
      <!-- #include file = "DataStore.asp" -->
      <html>
      <head>
      <title>Doctor</title>
      </head>
      <body>
      <p align="center"><font face="Times New Roman" size="5"><font
color="#006600">Doctor-Patient Communication Tool(DPCT)</font><span
style="background-color: #0000FF"><br>
      </span></font>
      </p>
      <%
         Dim rsUsers, rsUsers2, rsUsers3, rsUsers4, objConn, objComm,
ID, recordID, strSQL
         set objConn = Server.CreateObject("ADODB.Connection")
         set objComm = Server.CreateObject ("ADODB.Command")
         set rsUsers = Server.CreateObject("ADODB.Recordset")
         set rsUsers2 = Server.CreateObject("ADODB.Recordset")
         set rsUsers4 = Server.CreateObject("ADODB.Recordset")
         ID = Request.QueryString("Id")
         recordID = Request.QueryString("recordID")
         objConn.Open strConnect
         rsUsers.Open "select * from user where userID = " & ID,
objConn, adOpenForwardOnly, adLockOptimistic, adCmdText
         rsUsers2.Open "diagnosis2", objConn, adOpenForwardOnly,
adLockOptimistic, adCmdTable
         rsUsers4.Open "select diagnosis from symptom2 where UserID
like " & ID & " and testTime like '" & recordID & "'", objConn,
adOpenForwardOnly, adLockOptimistic, adCmdText
```

```
%>
<% curTime = now %>
<table align = "right">
 <tr>
   <td  width="198">
   <%= curTime %>
   </td></tr>
</table> <br>
<center>  <blockquote>
<table border="1" width="89%" bgcolor = "#EEffff">
  <tr>
    <td width="12%">Name </td>
    <td width="28%"><%=rsUsers("FirstName") & " " &
rsUsers("LastName") %></td>
    <td width="10%">Age</td>
    <td width="10%"><%= rsUsers("Age") %></td>
    <td width="5%">Gender</td>
    <td width="5%"><%= rsUsers("Gender") %></td>
    <td width="5%">Tel:</td>
    <td width="25%"><%= rsUsers("PhoneNo") %></td>  </tr>
  <tr>
    <td width="12%">Address</td>
    <td width="88%" colspan="7"><%= -+rsUsers("StreetAddress1")
& " "& rsUsers("City") &", "&rsUsers("state")&" "& rsUsers("Country")
%></td>  </tr>
</table> </blockquote></center> <p>
<blockquote><blockquote> <blockquote>
<center> <table><tr><td>
<img src = "D://inetpub/ftproot/<%= ID%>/<%= recordID %>.bmp"
width= "600" height="300"> </td><td> </td></tr>
<tr> <td align = right >
<% If session("Status") = "d" or session("Status") = "D" then
 Response.Write "<FORM NAME=DeleteDiagnosis ACTION=" & "'" &
"deleteDiagnosis.asp?Id=" & ID & "&deleteID=" & recordID & "'" & "
VALUE=deleteDiagnosis" & " METHOD=POST>"
 Response.Write "<INPUT TYPE=SUBMIT VALUE=DeleteDiagnosis
id=SUBMIT1 name=SUBMIT1> </FORM> "
 End if
```

```
%> </td> </tr></table> </center>
</blockquote></blockquote></blockquote>
Automatic Diagnosis Result:<br>
<% Dim strFile, objSearch, str
set objSearch = Server.CreateObject ("searchDiagData.searchData1")
strFile = "D:\Inetpub\ftproot\" & ID & "\" & recordID &  ".txt"
Response.write objSearch.getDiagnosis  (CStr (strFile ))  %>  <br>
<form name = "communication" method = "post"  action =
"phDiagnosisUpdate.asp?Id=<%=ID%>&recordID=<%=recordID%>">
<% objComm.ActiveConnection = strConnect
    strSQL = "select symptom  from symptom2 where UserID like " & ID
    strSQL = strSQL & " and  testTime like  '" & recordID
    strSQL = strSQL & "'"
    strSQL = strSQL & ";"
    objComm.CommandText = strSQL
    objComm.CommandType = adCmdText
    set rsUsers3 = objComm.Execute
    set objComm = nothing %>
    Patient <%=rsUsers("FirstName") & " " & rsUsers("LastName") %>'s
symptom is: <br>
<%= rsUsers3("symptom") %>
<%= rsUsers4("diagnosis") %> <br><br><br>
Dear Physician <i><%= session("FirstName") & " " & Session("LastName"
) %></i>, Please enter your detailed diagnosis results here:
<textarea rows="3" name="physicianDiagnosis" cols="70" ><%=
rsUsers4("diagnosis")%></textarea>     <p>
      <input type="Submit" value="Submit" name="SubmitDiagnosis">
      <input type="Reset" value="Reset" name="ResetDiagnosis"></p>
</form>
<% rsUsers.Close
   rsUsers2.Close
   rsUsers4.Close
%>
<br><a href="subHomepage.asp">BackHomepage</a>
<!-- #include file = "note.asp" -->
</body>
</html>
```

```
' ********************************************************************'
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: footer.asp
'Programmer: Hongwei Bi
'Date: 04/18/02
'Description: This is a footer file used as SSI.
'********************************************************************'
      <HTML>
      <HEAD>
      <META NAME="GENERATOR" Content="Microsoft FrontPage 4.0">
      <TITLE>Note.asp</TITLE>
      </HEAD>
      <BODY>
      <p vlign="bottom">
         <font size="3" >
            <hr>
            Copyright&copy; 2002 Global Charity, Inc., All rights
reserved.
         </font>
      </BODY>
      </HTML>
```

```
'*********************************************************************'
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: head1.asp
'Programmer: Hongwei Bi
'Date: 04/08/02
'Description: This a header file used in SSI.
'*********************************************************************'
      <HTML>
      <HEAD>
      <META NAME="GENERATOR" Content="Microsoft FrontPage 4.0">
      <TITLE>Head1</TITLE>
      </HEAD>
      <BODY>
      <p align="center"><font face="Times New Roman, Times, serif"
size="5" color="#008000">Global Charity Inc.
        </font><br>
        <font size="2">Tel: 909-888-6444, Fax: 909-800-2837 Email
info@GlobalCharity.us</font><br>
      </p>
      <hr>
      </BODY>
      </HTML>
```

```
'*********************************************************************
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: login.asp
'Programmer: Hongwei Bi
'Date: 04/18/02
'Date: Description: This file needs patients to enter 'their Email
address and password to logon the DPCT 'system.
'*********************************************************************
      <html>
      <head>
      <title>Login.asp</title>
      <meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
      </head>
      <body bgcolor="FFFFFFe" text="#000000" onload =
"document.forms.form1.Email.focus()">
      <!-- #include file="head1.asp" -->
      <div align="center"><font size="5" color="#004040"><br>
        Welcom to Doctor-Patient Communication Tool (DPCT)
</font><font size="5" color="#004040"><br>
        -- <font size="4">RYODORAKU Application on the Web
</font></font> </div>
      <font size="4"><br>
        <br>
        <%
         if Request ("SecondTry") = "True" then
            if request ("NofFount")= "True" then
               Response.Write "Please reenter your<i> Email
address</i> and<i> password</i> to login the System."
            else if Request ("WrongPW") = "True" then
               Response.write "Invalid Password. Please try again! "
            else
               Response.write "E-Mail Address not found. Please try
again!"
            end if
            end if
         else
```

```
            response.write "Please enter your <i> Email address</i>
and<i> password</i> to login the System."
          end if %>
          <P>
          </p>
        </font>
        <FORM ACTION="checkLogin.asp
             <% If Request("SecondTry")="True" Then
                  Response.Write "?SecondTry=True"
           End If %>"
           METHOD="POST" id=form1 name=form1>
          <TABLE BORDER=0 width ="50%">
            <TR>
              <TD>E-Mail Address:</TD>
               <TD><INPUT TYPE="Text" NAME="Email"
                 <% If Request("SecondTry") = "True" then %>
                   VALUE="<%=Session("EmailAddress") %>"
                 <% End If %>
                 SIZE="40"></TD>  </TR>
           <TR>
              <TD>Password:  </TD>
              <TD><INPUT TYPE="Password" NAME="Password"
SIZE="40"></TD> </TR>
           <TR>
              <TD></TD>
              <TD align=center><INPUT TYPE="Submit" VALUE="Login"
id=Submit1 name=Submit1>
                 
              <INPUT TYPE="RESET" id=RESET1 name=RESET1></TD></TR>
          </TABLE>
      </FORM>
      <p> </p>
      <p> </p>
      <!-- #include file = "note.asp" -->
      </body>
      </html>
```

```
'*********************************************************************'
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: patientPage.asp
'Programmer: Hongwei Bi
'Date: 04/18/02
'Description: After the Patient login the DPCT system, 'this 'age shows
up. The patient can update his/her
'personal information or search case history or check the 'nstant
diagnosis result.
'*********************************************************************'
      <HTML>
      <HEAD>
      <META NAME="GENERATOR" Content="Microsoft FrontPage 4.0">
      <TITLE></TITLE>
      </HEAD>
      <BODY><!-- #include file ="head2.asp" -->
       <font size = 4 >
         <blockquote>
         Dear Friend: <i> <%= session("FirstName")%></i><br>
       </font> <font size = 5>
        <CENTER> Welcome to DPCT system!<br><br> </font>
        <font size = 4>
         You can click the
        <A href="instantDiagnosis.asp"> Diagnosis </A>
         link to process diagnosis. Or search diagnosis
case histories. You also can update your personal information.<br>
         <A href="instantDiagnosis.asp" >Diagnosis </A>
         <A href="searchPatientHistory.asp" >Search Case History</A>
         <A href ="updatePatientInfo.asp"> Update Personal
Information </a>
         </CENTER>
      <p> </p> <p> </p>
      <p><a href="subHomepage.asp"><font size =
3>BackHomepage</font></a><br>
      <!-- #include file ="note.asp" --></blockquote></font>
      </BODY>
      </HTML>
```

```
'*********************************************************************'
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: strToTable.asp
'Programmer: Hongwei Bi
'Date: 04/18/02
'Description: This is a pure ASP page and as a SSI file. 'It 'onverts
the required string to table.
'*********************************************************************'
      <%
         Function strToTable (objRS)
            Dim strT
            Dim curFild
            strT = "<table border=0 cellspacing =5  cellpadding =2
><tr align = center>"
            For each curFild in rsUsers.Fields
               strT = strT & "<TD><strong>" & curFild.Name &
"</strong></TD>"
            Next
            strT = strT & "</TR>"
           while not rsUsers.eof
               strT = strT & "<TR aglign =center>"
               for each curFild in rsUsers.Fields
                  strT= strT & "<TD>" & curFild.Value & "</TD>"
               Next
               strT = strT &  "<TD> <a href= " & "' " &
rsUsers("EmailAddress") & " '" & ">" & "Records" & "</TD>"
            strT = strT & "</TR>"
               rsUsers.MoveNext
            Wend
            strT = strT & "</table>"
            strToTable = strT
         end Function
      %>
```

```
'*********************************************************************
'Project: Doctor-Patient Communication Tool(DPCT)
'Program: UpdateInfo.asp
'Programmer: Hongwei Bi
'Date: 04/28/02
'Description: This page used to update patients' personal
'information.
'*********************************************************************
<!-- #include file = "Datastore.asp" -->
<%
Dim rsUsers, objConn, loginStatus, objComm, strSQL, intNoOfRecords,
id
set objConn = Server.CreateObject("ADODB.Connection")
set objComm = Server.CreateObject ("ADODB.Command")
set rsUsers = Server.CreateObject("ADODB.Recordset")
objConn.Open strConnect
strSQL2 = "Select * from User " & _
          "where EmailAddress = '" & session("EmailAddress") & "';"
rsUsers.Open strSQL2, objConn, adOpenForwardOnly, adLockReadOnly,
adCmdText
rsUsers.Open  "User", objConn, adKeyset, adLockOptimistic, adCmdTable
strSQL = "Update User set FirstName = '" & Request.Form
("FirstName")& "'," & " LastName = '" & Request.Form("LastName")&"',"
 strSQL = strSQL &  " EmailAddress= '" & CStr(Request.Form("Email"))
& "', " & " Age = " & Request.Form ("Age") & ","
 strSQL = strSQL & "Password = '" & Request.Form("Password") & "', "
 &  " Gender  =  '" & Request.Form("Gender") & "',"
 strSQL = strSQL & "StreetAddress1 = '" &
Request.Form("StreetAddress1") & "',"  & "StreetAddress2 = '" &
Request.Form("StreetAddress2") & "',"
```

```
 strSQL = strSQL & "City = '" & Request.Form("City") & "'," & "State
= '" & Request.Form("State") & "',"
 strSQL = strSQL & "PostalCode = '" & Request.Form("PostalCode") &
"'," & "Country = '" & Request.Form("Country") & "',"
 strSQL = strSQl & "PhoneNo = '" &  Request.Form("PhoneNo") & "'" & "
where UserID = " & session("userID")
objComm.ActiveConnection = strConnect
objComm.CommandText = strSQL
objComm.CommandType = adCmdText
objComm.Execute intNoOfRecords
set rsUsers = Nothing
objConn.Close
set objConn = nothing
set objComm = nothing
        if session("status") = "d" or session("status") = "D" then
Response.Redirect "confirmation.asp"
        else Response.redirect "confirmationP.asp"
        End if
Response.Write "Privilege deny!"
%>
```

APPENDIX B

GLOSSARY OF TERMS

Ryodoraku	A theory used to diagnose people's health situation according to people's 12 meridians. "Ryodoraku" (ryo = good, do` is (electro) conductive, raku = line).
IIS	Internet Information Server. The Web server software included with Microsoft Windows NT. Supports applications that use CGI, SP, IDC and ISAPI; and interfaces with Windows NT and other services running on the server machine.
ODBC	Open Database Connectivity. An open standard originally developed by Microsoft to allow transparent data access to all kinds of data stores such as relational databases. Drivers are manufactured by third parties to suit their own data store
ASP	Active server pages is a technology that allows for the programmatic construction of HTML pages just before they are delivered to the browser. Producing dynamic, interactive Web application.
SSL	Secure Socket Layer. A technology originally developed by Netscape to provide client and server verification, and secure communication between a Web browser and server. Uses public key and secret key encryption.

GUI	Graphic User Interface
MDI	Multiple Document Interface
TCM	Traditional Chinese Medicine

REFERENCES

[1] Ryodoraku Treatment, An Objective Approach to Acupuncture; Masayoshi Hyodo; Japan Ryodoraku, Autonomic Nerve Society; 1975

[2] Beginning ASP 3.0; Chris Uiiman, David Buser, et.al; Wrox Press Ltd. 1999

[3] Using Active Server Pages; Scot Johnson, Keith Ballinger, et al.; QUE; 2000

[4] Visual Basic for Windows; Harold Davis; Peachpit Press; 2000

[5] The Revolutionary Guide to Visual Basic 4 professional; Larry Roof; Wrox Press Ltd.; 1996

[6] Visual Basic 5 Control Creation Starter Kit; Loren Eidahl, Michael McMillan; QUE; 1997

[7] Programming in Visual Basic 6.0; Julia Case Bradley, Anita C. Millspaugh; McGraw-Hill; 1999

[8] http://www.msdn.com

[9] http://chinesemedicinesampler.com

[10] Windows 2000 Programming; Herbert Schildt; McGraw-Hill; 2000

[11] http://www.pslgroup.com/dg/1fe786.htm

[12] http://www.healinx.com/Corporate/v/index.asp

[13] http://www.mysql.com/downloads/mysql-3.23.html

[14] "IEEE Recommended Practice for Software Requirements Specifications (IEEE Std 830-1998)"

[15] http://www.skylarkdevice.com